

NAVAL POSTGRADUATE SCHOOL

Monterey, California



THESIS

SHIPBOARD WIRELESS NETWORK APPLICATIONS

by

Tung T. Ly

June 2001

Thesis Advisor:
Second Reader:

Xiaoping Yun
John C. McEachen

Approved for public release; distribution is unlimited

Form SF298 Citation Data

Report Date <i>("DD MON YYYY")</i> 15 Jun 2001	Report Type N/A	Dates Covered (from... to) <i>("DD MON YYYY")</i>
Title and Subtitle SHIPBOARD WIRELESS NETWORK APPLICATIONS		Contract or Grant Number
		Program Element Number
Authors		Project Number
		Task Number
		Work Unit Number
Performing Organization Name(s) and Address(es) Naval Postgraduate School Monterey, CA 93943-5138		Performing Organization Number(s)
Sponsoring/Monitoring Agency Name(s) and Address(es)		Monitoring Agency Acronym
		Monitoring Agency Report Number(s)
Distribution/Availability Statement Approved for public release, distribution unlimited		
Supplementary Notes		
Abstract		
Subject Terms		
Document Classification unclassified		Classification of SF298 unclassified
Classification of Abstract unclassified		Limitation of Abstract unlimited
Number of Pages 125		

SHIPBOARD WIRELESS NETWORK APPLICATIONS

Tung T. Ly-Lieutenant, United States Coast Guard

B.S., Virginia Tech, 1991

Master of Science in Electrical Engineering–June 2001

Advisor: Xiaoping Yun, Department of Electrical and Computer Engineering

Second Reader: John C. McEachen, Department of Electrical and Computer Engineering

Recently, the need to leverage technologies for better utilizing valuable personnel resources has become more important. Wireless Local Area Networks (WLANs) have been shown to be an enabling technology that allows companies in commercial industry to become more productive. Research has been conducted at the Naval Postgraduate School to determine how this technology can be utilized to help the Navy perform shipboard operations more efficiently.

Continuing the work of previous theses at NPS, the objective of this thesis is threefold. First, WLAN standards are examined. Second, laboratory tests are conducted to determine the performance of WLANs in which access points are configured as radio repeaters. Finally, a web-based application is developed for shipboard gage calibrations. The application automates major portion of gage calibration process by allowing technicians to submit and to view the calibration results using a web browser through wired or wireless LANs.

Testing results show that the access points from certain vendors are able to operate as radio repeaters and still provide adequate performance. Repeater functionality is not specified in IEEE 801.11 standards, and its implementation is vendor specific. Demonstration of the web-based gage calibration application shows that it is effective in improving calibration efficiency.

DoD KEY TECHNOLOGY AREA: Computing and Software

KEYWORDS: IEEE 802.11, Wireless Local Area Network, Active Server Pages, Internet Database

REPORT DOCUMENTATION PAGE			<i>Form Approved OMB No. 0704-0188</i>	
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instruction, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188) Washington DC 20503.				
1. AGENCY USE ONLY (Leave blank)		2. REPORT DATE June 2001	3. REPORT TYPE AND DATES COVERED Master's Thesis	
4. TITLE AND SUBTITLE: Shipboard Wireless Network Applications			5. FUNDING NUMBERS	
6. AUTHOR(S) Tung T. Ly				
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School Monterey, CA 93943-5000			8. PERFORMING ORGANIZATION REPORT NUMBER	
9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES) N/A			10. SPONSORING / MONITORING AGENCY REPORT NUMBER	
11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.				
12a. DISTRIBUTION / AVAILABILITY STATEMENT Approve for public release; distribution is unlimited			12b. DISTRIBUTION CODE	
13. ABSTRACT (maximum 200 words) <p>Recently, the need to leverage technologies for better utilizing valuable personnel resources has become more important. Wireless Local Area Networks (WLANs) have been shown to be an enabling technology that allows companies in commercial industry to become more productive. Research has been conducted at the Naval Postgraduate School to determine how this technology can be utilized to help the Navy perform shipboard operations more efficiently.</p> <p>Continuing the work of previous theses at NPS, the objective of this thesis is threefold. First, WLAN standards are examined. Second, laboratory tests are conducted to determine the performance of WLANs in which access points are configured as radio repeaters. Finally, a web-based application is developed for shipboard gage calibrations. The application automates major portion of gage calibration process by allowing technicians to submit and to view the calibration results using a web browser through wired or wireless LANs.</p> <p>Testing results show that the access points from certain vendors are able to operate as radio repeaters and still provide adequate performance. Repeater functionality is not specified in IEEE 801.11 standards, and its implementation is vendor specific. Demonstration of the web-based gage calibration application shows that it is effective in improving calibration efficiency.</p>				
14. SUBJECT TERMS IEEE 802.11, Wireless Local Area Network, Active Server Pages, Internet Database			15. NUMBER OF PAGES 127	
			16. PRICE CODE	
17. SECURITY CLASSIFICATION OF REPORT Unclassified	18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified	19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified	20. LIMITATION OF ABSTRACT UL	

THIS PAGE INTENTIONALLY LEFT BLANK

Approved for public release; distribution is unlimited

SHIPBOARD WIRELESS NETWORK APPLICATIONS

Tung T. Ly
Lieutenant, United States Coast Guard
B.S., Virginia Tech, 1991

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN ELECTRICAL ENGINEERING

from the

**NAVAL POSTGRADUATE SCHOOL
June 2001**

Author: _____
Tung T. Ly

Approved by: _____
Xiaoping Yun, Thesis Advisor

John McEachen, Second Reader

Jeffrey Knorr, Chairman
Department of Electrical and Computer Engineering

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

Recently, the need to leverage technologies for better utilizing valuable personnel resources has become more important. Wireless Local Area Networks (WLANs) have been shown to be an enabling technology that allows companies in commercial industry to become more productive. Research has been conducted at the Naval Postgraduate School to determine how this technology can be utilized to help the Navy perform shipboard operations more efficiently.

Continuing the work of previous theses at NPS, the objective of this thesis is threefold. First, WLAN standards are examined. Second, laboratory tests are conducted to determine the performance of WLANs in which access points are configured as radio repeaters. Finally, a web-based application is developed for shipboard gage calibrations. The application automates major portion of gage calibration process by allowing technicians to submit and to view the calibration results using a web browser through wired or wireless LANs.

Testing results show that the access points from certain vendors are able to operate as radio repeaters and still provide adequate performance. Repeater functionality is not specified in IEEE 801.11 standards, and its implementation is vendor specific. Demonstration of the web-based gage calibration application shows that it is effective in improving calibration efficiency.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

I.	INTRODUCTION.....	1
A.	COMPUTER NETWORKS.....	2
1.	History of Computer Networks.....	2
2.	Computer Connectivity.....	3
3.	Wireless LAN Definition.....	3
a.	<i>Infrared as a Medium</i>	4
b.	<i>Laser as a Medium</i>	6
c.	<i>Radio Frequency as a Medium</i>	7
B.	WIRELESS LAN BENEFITS.....	8
1.	Non-military Applications of WLAN	8
a.	<i>Manufacturing</i>	8
b.	<i>Education</i>	9
c.	<i>Retail Sale</i>	9
d.	<i>Difficult to Install Locations</i>	10
e.	<i>Security</i>	10
f.	<i>Conferences and Meetings</i>	10
2.	Military Applications	11
3.	History of 802.11 and Wireless LAN	11
C.	THESIS GOAL AND ORGANIZATION.....	13
II.	BACKGROUND.....	15
A.	RADIOWAVE	15
1.	RF Propagation	15
2.	Path Loss	16
3.	Multipath.....	18
4.	Dealing With Multipath Interference.....	19
B.	SPREAD SPECTRUM TECHNOLOGY	21
C.	ORTHOGONAL FREQUENCY DIVISION MULTIPLEXING (OFDM).....	22
D.	OPEN SYSTEM INTERCONNECT (OSI) LAYERS.....	23
1.	IEEE 802.11a physical layer.....	25
a.	<i>Physical Layer Convergence Procedure Protocol Data Unit</i>	26
b.	<i>Physical Medium Dependent (PMD)</i>	27
c.	<i>IEEE 802.11a Frequency and Regulations</i>	28
2.	802.11a Medium Access Control (MAC) Layer	30
a.	<i>Control Access to Wireless Medium</i>	31
b.	<i>Provide Reliable Data Delivery Service</i>	32
E.	WIRELESS SECURITY	33
1.	Wire Equivalent Privacy (WEP).....	33
2.	Standards Security Precautions.....	34
F.	WIRELESS LAN ISSUES.....	35

III.	TESTING OF WLAN COMPONENTS	37
A.	SELECTION OF EQUIPMENT FOR TESTING	37
B.	EQUIPMENT USED TO SUPPORT TESTING	38
C.	SOFTWARE USED FOR EVALUATION.....	39
	1. Cisco Utilities	39
	2. Lucent WaveLAN Utilities	41
	3. WS_Ping ProPack	42
D.	EQUIPMENT TESTED	44
	1. Cisco 340 and Lucent WaveLAN Gold PC Card Client Adaptors.....	45
	2. Aironet 4800, Cisco 340, and Cisco 350 Access Points.....	45
E.	BASELINE TESTING.....	46
F.	ONE ACCESS POINT.....	49
G.	ONE RADIO REPEATER.....	50
H.	TWO RADIO REPEATERS.....	52
I.	DISCUSSION OF RESULTS.....	54
IV.	WLAN APPLICATIONS	57
A.	GPTE SEMI-AUTOMATED HIGH PRESSURE CALIBRATOR.....	57
B.	WIRELESS LAN AND THE GSAHPC SYSTEM	58
C.	INTERNET AND DATABASES	59
	1. IDS Configuration	60
	2. Submission of Data.....	61
	3. IDS Server Configuration.....	62
	4. Running IDS Inside Microsoft's Internet Information Services ..	62
	5. Client Java Application	63
	6. Alternatives to Java Applications on the Client Machine	64
	a. Using Java Applets.....	65
	b. HTML Extensions.....	65
	c. JavaServer Pages (JSP)	66
	d. Active Server Page (ASP).....	66
	7. Discussion of Alternative Selection.....	67
V.	CONCLUSIONS	69
A.	TECHNOLOGIES AND STANDARDS USED IN WLAN.....	69
B.	TESTING OF WLAN CONFIGURATIONS.....	70
C.	DEVELOPMENT OF A WLAN APPLICATION TO COLLECT GAGE CALIBRATION	70
D.	RECOMMENDATIONS FOR CONTINUED RESEARCH.....	71
	1. Future WLAN Studies	71
	2. Future WLAN Applications	72
	APPENDIX A – JAVA APPLICATION CODE TO SUBMIT DATA	73
	APPENDIX B – HTML CODE TO VIEW DATA.....	78
	APPENDIX C – VIEWDATA JAVA APPLLET	80

APPENDIX D – HTX CODE TO VIEW DATA.....	87
APPENDIX E – ASP CODE TO VIEW DATA.....	89
APPENDIX F – ASP JAVASCRIPT CODE TO SUBMIT DATA	91
APPENDIX G – SUBMITFILESRIPT.ASP.....	93
LIST OF REFERENCES	97
INITIAL DISTRIBUTION LIST	101

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF FIGURES

Figure 1.2 – AEI Laser System From Ref. [4].....	6
Figure 2.1 – Components of Radio Wave Propagation.....	16
Figure 2.2 – Foot Pattern For Omni Directional and Directional Antenna From Ref. [14]....	18
Figure 2.3 – Frequency Division Multiplexing.....	22
Figure 2.4 – Orthogonal Frequency Division Multiplexing (OFDM)	23
Figure 2.5 – OSI Layers and IEEE 802.11a.....	25
Figure 2.6 – PPDU Frame Format From Ref. [9]	26
Figure 2.7 – PMD Transmitter and Receiver Functional Block Diagram After Ref. [9].....	28
Figure 2.8 – OFDM Frequency Channels From Ref. [9].....	29
Figure 2.9 – MAC Frame after Ref. [22]	32
Figure 2.10 – Parking Lot Attack.....	33
Figure 3.1 – WLAN With Access Point as Radio Repeater.....	37
Figure 3.2 – Linksys 10/100 Integrated PC Card From Ref. [28].....	39
Figure 3.3 – 3Com OfficeConnect Dual Speed Hub 8 From Ref. [29]	39
Figure 3.4 – Cisco Link Status Meter Program.....	40
Figure 3.5 – Cisco 340 Access Point Association Table.....	41
Figure 3.6 – WaveLAN’s WaveMANAGER/CLIENT Program	42
Figure 3.7 – WS_Ping ProPack Throughput Measurement	43
Figure 3.8 – Cisco 340 and WaveLAN Card from Ref. [30].....	45
Figure 3.9 – Aironet 4800, Cisco 340 and Cisco 350 Access Points.....	46
Figure 3.10 – Baseline Testing Configuration	46

Figure 3.11 – One Access Point to Wireless Client.....	49
Figure 3.12 – One Repeater Wireless Network.....	51
Figure 3.14 – Two Repeaters Network Results for Cisco PC Card	54
Figure 4.1 – GSAHPC System From Ref. [31].....	58
Figure 4.2 – Connectivity Between JDBC and Database.....	60
Figure 4.3 – IDS Configuration From Ref. [32]	61
Figure 4.4 – File Submission Java Application	64

LIST OF TABLES

Table 3.1 – Baseline Testing with Linksys 10/100 PC Card	48
Table 3.2 – One Access Point Results	50
Table 3.3 – One Radio Repeater Accessing Through Repeater	51
Table 3.4 – One Radio Repeater Accessing Through Access Point	52
Table 3.5 – Two Repeaters Accessing Through Second Repeater	53

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF SYMBOLS, ACRONYMS AND/OR ABBREVIATIONS

ARPA	Advance Research Project Agency
ASP	Active Server Pages
BPSK	Binary Phase Shift Keying
CFR	Code of Federal Regulations
COTS	Commercial Of The Shelf
DCF	Distributed Coordination Function
DQFSK	Differential Quadrature Phase Shift Keying
DSSS	Direct Sequence Spread Spectrum
ETSI	European Telecommunication Standardization Institute
FDM	Frequency Division Multiplexing
FHSS	Frequency Hopping Spread Spectrum
GPTE	General Purpose Test Equipment
HIPERLAN	High Performance Radio LAN
IEEE	Institute of Electrical and Electronics Engineers
IR	Infra Red
IrDA	Infra Red Data Association
IRLAN	Infra Red LAN
ISM	Industrial Scientific Medial
JSP	JavaServer Page
OFDM	Orthogonal Frequency Division Multiplexing
OSI	Open System Interconnect

PCF	Point Coordination Function
PLCP	Physical Layer Convergence Procedure
PMD	Physical Medium Dependent
PSDU	Physical sub-layer Service Data Unit
MAC	Medium Access Control
MPCU	Medium Access Control Protocol Data Unit
PLCP	Physical Layer Convergence Procedure
PMD	Physical Medium Dependent
PPDU	PLCP Protocol Data Unit
PSDU	Physical sublayer Service Data Unit
RADIUS	Remote Access Dial-In User Service
RF	Radio Frequency
SAP	Service Access Point
SSID	Service Set Identifier
U-NII	Unlicensed National Information Infrastructure
WAN	Wide Area Network
WEP	Wire Equivalent Privacy
WLAN	Wireless Local Area Network

ACKNOWLEDGMENTS

I would like to thank Professor Xiaoping Yun and John McEachen for their guidance and patience. Additionally, I would like to thank my mother and Trinh for their encouragement.

THIS PAGE INTENTIONALLY LEFT BLANK

DISCLAIMER STATEMENT

The selection of the particular hardware and software discussed in this thesis does not constitute an official endorsement of its use in any particular application.

THIS PAGE INTENTIONALLY LEFT BLANK

EXECUTIVE SUMMARY

As the Navy moves into the Twenty-First Century, the need to leverage technologies for better utilizing valuable personnel resources has become more important. Wireless Local Area Networks (WLANs) have shown to be an enabling technology that allows companies in the commercial industries to become more productive. Research has been conducted at the Naval Postgraduate School (NPS) to determine how this technology can be utilized to help the Navy perform shipboard operations more efficiently.

Continuing the work of previous theses in the WLAN research, the objective of this thesis is threefold. First, WLAN standards, in particular the latest IEEE 802.11a standard, are examined. Second, laboratory tests are conducted to determine the performance of WLANs in which access points are configured as radio repeaters. The purpose of repeaters is to extend the range of WLANs. Finally, a web-based application is developed for shipboard gage calibrations.

The technology used by the IEEE 802.11a standard was analyzed and compared with the 802.11b and the European's High Performance Radio Local Area Network 2 (HIPERLAN 2) standard. The difference between the IEEE 802.11a standard and the IEEE 802.11b standard is in the physical layer of the Open System Interconnect (OSI) layers. In the physical layer, the IEEE 802.11a standard uses Coded Orthogonal Frequency Division Multiplexing (COFDM) and can achieve a throughput of up to 54 Mbps. This compares with Spread Spectrum (SS) used by the IEEE 802.11b standard, which can only achieve a throughput of 11 Mbps. Both the IEEE 802.11a standard and

the HIPERLAN 2 standard operate in the 5 GHz frequency range and use COFDM. However, because these standards are implemented according to local regulations, they are not compatible. There are working groups attempting to unify these two standards to form a new compatible standard.

Testing of the 802.11b access points shows that access points from certain vendors are able to operate as radio repeaters. Because the repeater function is not specified in the IEEE 802.11 standards, not all vendors offer this capability. Additionally, implementations of this feature are vendor specific therefore access points and radio repeaters among different vendors may not interoperable. Tests on three models of the Cisco access points show that the average throughput for a one radio repeater system is 2.97 Mbps. For a two repeaters system, an average throughput of 821 Kbps is measured. These throughputs are sufficient for many applications including the gage calibration application developed in this thesis.

A web-based application is developed for shipboard gage calibrations. The application automates major portion of the gage calibration process by allowing technicians to submit calibration results to a database and to view the results using a web browser through the wired or wireless LANs. Active Server Pages (ASP), Java application, and Java applet technologies along with several database tools are used to develop the application.

I. INTRODUCTION

Since the invention of computers, computing costs have decreased while computing power has increased at a steady pace. This pace has followed Moore's Law, where computing power has roughly doubled every eighteen months. [Ref. 1] This rapid increase in computer's abilities and their decreasing cost has made it an indispensable tool in today's businesses. Every business needs computers to remain competitive, which includes the military. The cost of computer and information technologies is minimal compared to the cost of personnel. With the end of the cold war and reduced defense spending, having unlimited personnel resources is no longer an option. The Navy must use its limited number of personnel most efficiently. One of the tools that can help the Navy leverage personnel, its most valuable resource, is to use computers effectively. This is essential to achieve the military goal of working smarter not harder.

This thesis' objective is to help the Navy implement wireless local area network (WLAN) technology by completing the following tasks:

- Examine WLAN standards and technologies
- Test WLAN configurations that may be used in a ship deployment
- Develop a WLAN application for shipboard gage calibrations

Other current thesis work related to this thesis at NPS include studying the electromagnetic compatibility of wireless systems onboard Navy's ships, examining Bluetooth technology, and developing damage control software for submarines using WLANs and portable computers.

This thesis is organized into five chapters. Chapter I is the introduction. Chapter II will discuss the background of wireless LAN. Chapter III will present testing results of

wireless LAN hardware. Chapter IV will discuss applications using WLAN for the Navy. Finally, chapter V is the conclusion and recommendation for future research.

A. COMPUTER NETWORKS

In today's environment if a computer is not networked with other computers, it is almost useless. This is because the information that the computer processes is usually gathered by different computers. Once the computer processes the data, it usually has to transfer the results to other users on other computer systems. Computer networks have grown to be so important that it has generated an entire industry dedicated to it. This includes network security, fiber optics, data warehouse, remote client software, wired hardware and wireless hardware. There are even secondary industries that are built on computer network technology like distance learning, music file swapping, and e-commerce.

1. History of Computer Networks

It is fitting that the Navy should exploit wireless LAN (WLAN) and the Internet. Many of the technologies that make WLAN a reality came from Department of Defense (DoD) research. The Internet, which is the source for many of today's innovations in computer networks, started out as ARPANET (Advance Research Project Agency Network). ARPANET was initially developed so that government research institutions could pool scarce computing power for research. In 1982 ARPA specified that the TCP/IP suite would be used for ARPANET. Eventually, this network grew so much and with such a diverse interest that it was no longer supported by ARPA. [Ref. 1]

Xerox developed the Ethernet transmission protocol in 1973. In 1980, Xerox, Digital Equipment Company (DEC), and Intel jointly standardized and released the first

Ethernet specification. In 1983, the Institute of Electrical and Electronics Engineers (IEEE) 802.3 Working group released the first IEEE standard for Ethernet. Currently 10/100/1000 Mbps Ethernet are commonly used in most LANs. The IEEE 802.11 standard tries to build on this success by making it easy to integrate WLANs over an existing wire Ethernet system and provide the existing system with the added wireless capability option.

2. Computer Connectivity

A computer network is a network of computers, printers, data storage devices, modems, etc. When this network of computers is confined within a building or a campus it is called a Local Area Network (LAN). When larger geographical distances separate the resources of this network it is called a Wide Area Network (WAN). One of the main differences between a LAN and a WAN is that LAN connections will have higher bandwidth. This is because usually many people or applications share one WAN connection, while there is usually a dedicated connection for a WLAN (switched Ethernet).

3. Wireless LAN Definition

A wireless LAN is a LAN in which computers are connected together without using wires. There are two main mediums that can provide this non-wired connection. The first method is light, which can be further broken down into Infrared (IR) and laser. The other medium is radio frequency electromagnetic waves. A discussion of each medium including their pros and cons will follow.

a. *Infrared as a Medium*

Infrared (IR) is used in some short-range indoor wireless LANs. There are two types of IR systems in use, direct and diffused systems. Direct IR works similar to a docking station where users place the device in a docking station to connect that device to the network or another device. In a direct IR system, when two devices are to be connected, the user would aim the two IR transceivers in the general direction toward each other to establish the connection. An example of this system is the Clarinet System (see Figure 1.1 below). Even though the IEEE 802.11 standard has a specification for IR physical layer, at this time most of the direct IR WLAN products use the Infrared Data Association (IrDA) standards. The current IrDA standard supports up to an 8 Mbps data link and the IrDA is working on a 16 Mbps data link standard.

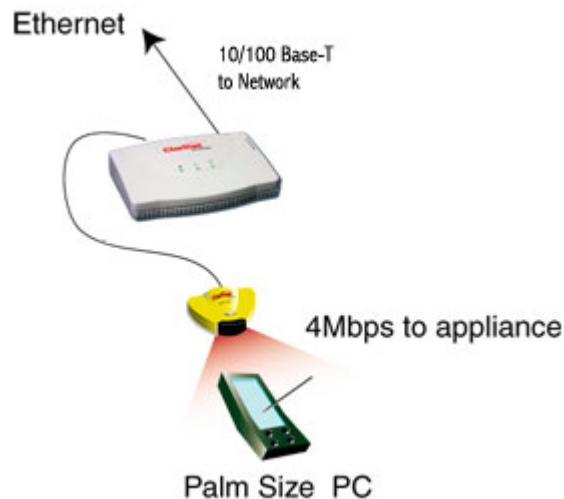


Figure 1.1 – Clarinet System Direct IR System From Ref. [2]

Diffused IR system allows users more flexibility in movement. A diffused IR system uses both wide-angle transmitters and receivers to communicate. IR signals can be reflected off surfaces like walls and ceilings if a direct line of site is not available. This frees the user from the requirement to maintain constant aim of the two devices

when communicating. Communication with diffused IR is usually limited to an area about 7 meters by 7 meters [Ref. 3].

Some of the advantages of IR WLAN systems are:

- The users do not have to be concerned with radio interference in a noisy radio environment.
- If used in a closed room, users do not have to be concerned with the signal being intercepted by hackers, because IR cannot penetrate walls like radio signals.
- The FCC does not regulate the frequencies in the IR range.
- IR WLAN is the cheapest WLAN alternative.

There are however some disadvantages to the IR WLAN system also. The IR WLAN system has a lower speed, its transmission range is more limited and it cannot penetrate walls like radio frequency waves. Because of this, an IR WLAN installation would require an access point or an access point repeater in every room or every area greater than 7 meter by 7 meter to give the system adequate coverage. However installing repeaters or access points that are physically connected by wires would negate the main benefit of a WLAN. Secondly, if the system is using the IrDA 1.1 FIR standard, it can only achieve a transmission speed of 4 Mbps. Some vendors like IRLAN and JVC-Victor offer products using their proprietary protocols for the IR portion of the system. These systems then use a well-established standard like 10 BaseT for the wired portion that the user would interface with. These devices offer 10 Mbps bandwidth with the IR transceivers placed up to 15 meters apart [Ref. 3]. But these systems are not designed for palm or mobile computers. They are designed to provide a wireless connection for desktop computers since these devices have to maintain IR aim.

b. Laser as a Medium

Laser is another wireless option. Usually laser is applied over longer distance and acts more like a wireless bridge than a WLAN. A wireless bridge is a device used to connect two LANs together. These systems employ two wireless bridge devices, one on each LAN to interconnect the two LANs. Because these bridges sometime provide links in a LAN backbone, the high speed of the laser WLAN may provide the best solution. AEI Communication is one of the companies offering such a product. They use up to four 100 mW laser diodes in each transceiver to create a point-to-point wireless connection up to 5000 meters apart and can support data rates up to 155 Mbps. They support most standard LAN protocols including Ethernet, Fast Ethernet, and ATM. With the AEI system the two remote networks will be interconnected as if they are connected directly by local network wires. Figure 1.2 below from AEI shows a typical network setup using their system. AEI claims that because of the narrow focus of the laser beam an unauthorized interception of the signal would require the interceptor to be very close to the direct path of the laser beam making unauthorized interception of signal difficult.

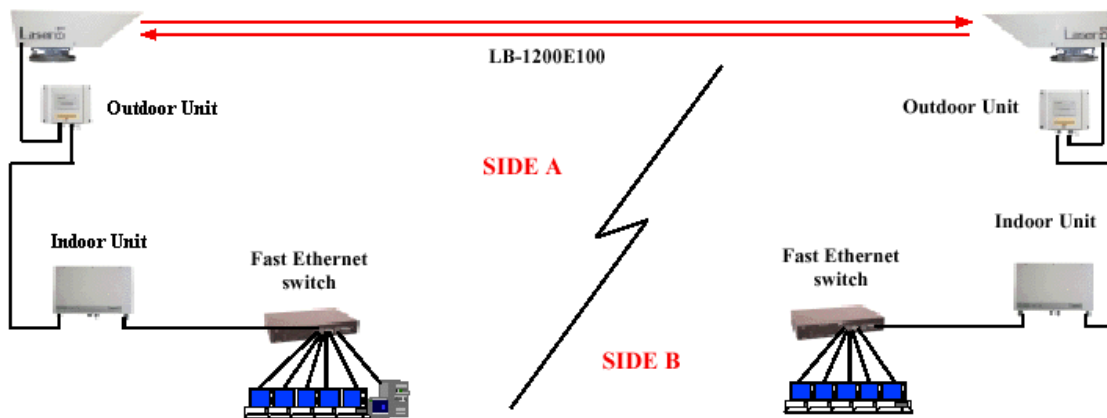


Figure 1.2 – AEI Laser System From Ref. [4]

Laser wireless systems as described above have very specific applications. It is only intended to be used as the last mile connection or to connect buildings within a campus or close to each other; 5000 meter is the upper limit distance for the AEI system. The main weakness of this system is that it could be affected by weather and sunlight. AEI recommends that the system be aligned in the north south direction to avoid the sunset and sunrise that could affect an east west aligned system [Ref. 4]. However, changing the location of two remote sites 5000 meter apart to get the ideal north south alignment may not be an option. Additionally, in foggy areas like San Francisco, the system will have too much down time due to fog to be acceptable.

c. Radio Frequency as a Medium

Currently, the most widely use medium for a WLAN is Radio Frequency (RF) Electromagnetic waves. There are many reasons why RF is a popular medium for a WLAN. RF systems can be applied over a short-distance link like IR WLANs. Bluetooth is a RF standard that is designed to accomplish this cheaply and is competitive with current IR WLAN (IrDA) devices. RF systems can also be used over long distances as a bridge to interconnect two LANs like the laser WLAN system. Cisco advertised that their product using 802.11b standard could reach up to a range of 25 miles at 2 Mbps or 11.5 miles at 11 Mbps [Ref. 5]. RF systems can also fill in the area of coverage that extend beyond the IR system, and cover distances of up to 100 meters indoor with free roaming capability.

There are several benefits that an RF system has over a light based system:

- Radio signals do not require a direct line of site. This allows users to have more freedom of movement while being connected than an IR connection.
- Radio signals can penetrate walls and ceiling.

- Radio signals are less prone to be affected by the weather than a laser system. This makes RF systems the only alternative for some installation.
- Most radio signal transceivers are omnidirectional. One transceiver in the middle of a room can cover the entire room.

B. WIRELESS LAN BENEFITS

There are many applications that are ideal for a WLAN. Basically any application that requires network resources but cannot be attached to a tether line is ideal for a WLAN. Areas where WLANs may provide the best solution are too numerous to list. Below are a small number of examples where a WLAN provides the best solution.

1. Non-military Applications of WLAN

Even though the non-military applications mentioned below are currently used in the commercial industry, the military employs many of these functions and could also benefit from using WLANs in a similar manner.

a. Manufacturing

Many of today's companies are using Ethernet to control automation instead of using proprietary protocols and standards. Currently, wired Ethernet is poised to take over the automation control industry where proprietary systems now in use cost from \$300 to \$900 and top out at 12Mbps [Ref. 6]. Using WLANs is the next extension to allow for greater mobility in manufacturing robots. A WLAN will allow machines to be more mobile so that they can adapt easily to a changing manufacturing process. Because of the time critical nature of control signals, high speed wired LANs are currently preferred. With the continuing improvement in WLAN technology, manufacturing processes will likely be able to use WLAN in the near future.

b. Education

Drexel University has installed a wireless network that not only allows the university to overcome its lack of computer lab space; it has made computer use more friendly for the students. Students now have the option to access the school network using a notebook computer equipped with a wireless LAN PC Card. This allows students to go anywhere on campus to perform their work.

The University of North Carolina at Wilmington uses a WLAN in classrooms to allow students to have more interaction with professors during instruction. It allows a professor to poll students to see instantly if the class understands the materials. This is done by students answering questions anonymously using their wireless-networked-laptop computers, which provides immediate feedback and allows the professor to assess the class understanding. The experiment shows that this medium resulted in a higher level of student participation as compared with traditional classroom interaction. This system also allows professors to administer quizzes and tests by using these WLAN computers. [Ref. 7]

c. Retail Sale

Electronic Boutique uses a WLAN to speed up customers' checkout. In some stores where they have a limited amount of checkout counters, Electronics Boutique has allowed sale clerks to walk around the store to help customers make their selection. When the customers are ready to make the purchase, the clerk can process a credit card purchase immediately using a palm computer connected by WLAN. Because of this they can eliminate customer wait at the check out counter and also reduce the chance of the customer changing their mind.

d. Difficult to Install Locations

Other situations where a WLAN is an ideal choice include wiring a historic building. This is a situation that the Coast Guard encounters regularly in trying to upgrade the Aids to Navigation Systems, many of which are located in historic lighthouses. Because of their historic designation it is very difficult to obtain permission to install new conduits or cable runs for networks. With a WLAN the Coast Guard can accomplish this without altering the buildings. Another ideal use for a WLAN is at home; most homes have not been built with provision to support a LAN. However, now with many people telecommuting and many homes having multiples computers, having a home network is common. A WLAN allows the homeowner to install a network quickly and inexpensively without having to pay for expensive contractors to install unsightly cables within their home.

e. Security

Security application is another area where a WLAN can be invaluable. A WLAN can be used to permit roving robots to maintain communications with a human guard, who can override a preprogrammed route to investigate suspicious areas as necessary. WLANs can also act as a backup to the wired LAN in security applications to give the system resiliency to tampering and failures.

f. Conferences and Meetings

WLANs have a set up option that is particularly well suited for conferences and meetings. A WLAN can be configured so that a client will be associated with the strongest signal. Users can also be isolated to a group with a common interest by giving each group a unique Service Set Identifier (SSID). With this configuration,

when a participant walks into a meeting room they can set their SSID and be connected to the rest of the participants in that room.

2. Military Applications

A WLAN can facilitate Coast Guard law enforcement boarding by providing officers with the ability to access their lookout database. Currently, larger Coast Guard cutters have connectivity to the law enforcement database. However, when a boarding team launches a small boat from the larger vessel to conduct boarding, they have to use voice communications to relay information back to the main cutter to run the database check. This voice communication is less secure and by transmitting without encryption, the person who a check is being made can have their personal data compromised if it is sent on a public marine radio channel. With a WLAN, the boarding team could carry a palm type computer with a direct link back to the main cutter to perform database check. Additionally, this computer can be used to enter boarding information that can be up-linked to the database, eliminating the need for a paper to computer entry task.

A WLAN will allow shipboard personnel to have mobile access to network computers to conduct their work in a more efficient and effective manner. The palm computer can also be used to send e-mails to family to boost morale as bandwidth permits. In Chapter IV, this thesis will examine the use of WLAN for gage calibrations and damage control in the Navy.

3. History of 802.11 and Wireless LAN

In 1970 Norman Abrahamson from the University of Hawaii developed a system to facilitate computer networking using a single radio channel. Many of the protocols used by Abrahamson were later adopted in the Ethernet standard [Ref. 8]. In 1995, the

FCC modified its Code of Federal Regulation, Title 47, Part 15 that regulates the Unlicensed National Information Structure (U-NII). This regulation allows use of unlicensed equipment to use certain frequencies contained within the bands 2.4GHz to 2.5 GHz and 5.1 GHz to 5.9 GHz. As long as the equipment meet the rules set by the FCC, the user can operate in these frequency bands without obtaining FCC license. This ability to use unlicensed equipment is what made IEEE 802.11 and many of the other WLAN feasible for many individuals and companies. In Chapter II this thesis will examine these frequencies and their international availability.

In June 1997, the IEEE 802.11 Working Group released the 802.11 standard providing up to 2Mbps bandwidth in the 2.4GHz to 2.5GHz frequency range. In 1999 the IEEE 802.11 Working Group released 802.11a and 802.11b standards. The 802.11b standard supports up to 11 Mbps bandwidth in the 2.4 GHz to 2.5 GHz range, while the 802.11a standard supports up to 54 Mbps using the 5 GHz frequency band [Ref. 9]. Because IEEE 802.11 currently dominates the market, this thesis will concentrate on this standard. Bluetooth and HomeRF are starting to have more products available commercially. However, they are not yet in wide use at this time. Developers are still working on products that use these standards. Another WLAN standard is High Performance Radio LAN (HIPERLAN) developed by the European Telecommunication Standardization Institute (ETSI). Currently, there is no HIPERLAN 2 product available and the future of HIPERLAN 1 seems to be uncertain. There are many prominent communications companies in the HIPERLAN 1 and HIPERLAN 2 membership however there is no product available at this time [Ref. 10].

C. THESIS GOAL AND ORGANIZATION

This is an application thesis. The main goal of this thesis is to help the Navy implement a shipboard WLAN system. To accomplish this goal this thesis will examine the current state of the markets for WLAN systems to determine which products best suit the Navy's needs. Previous theses at NPS have already started to examine this topic. However, because of the rapid changes in technology and market conditions this thesis will reevaluate any changes and determine how these changes may be applicable in current and future applications. Additionally, WLAN configurations that may be used during an actual installation will be tested. This thesis will also discuss how WLANs can be used in conducting shipboard gage calibrations. Issues concerning WLANs that utilize java programming and Internet database servers will be discussed. Software development and issues for implementing this system will be examined. In Chapter I of this thesis basic concepts, history, and example applications of WLANs were introduced. Chapter II will examine the technical details of a WLAN including concerns for implementation, competing standards, and RF properties. Chapter III will examine testing of WLAN components under different configurations. Chapter IV will go into detail of WLAN application in the Navy. The main focus will be on the gage calibration program with a short discussion of damage control application being examined by other theses at NPS. Finally, Chapter V will examine areas of future study and continued research in this field.

THIS PAGE INTENTIONALLY LEFT BLANK

II. BACKGROUND

This thesis will concentrate on the use of radio waves as the physical carrier of WLAN signals. The first section of this chapter will examine general properties of radio waves. The Open System Interconnect (OSI) model will be introduced and the thesis will go into details of the 802.11 standard and how it fits into the OSI model. This chapter will also examine details in the implementation of 802.11a physical layer, the 802.11 Medium Access Control (MAC) layer, and the Wired Equivalent Privacy feature of IEEE 802.11.

A. RADIOWAVE

This thesis will concentrate on the use of radio waves as the physical carrier of WLAN signals. The FCC defines the radio frequency spectrum as "the part of the natural spectrum of electromagnetic radiation lying between the frequency limits of 9 kilohertz and 300 gigahertz" [Ref. 11]. The protocols that will be examined in this thesis use frequencies either in the 2 gigahertz or the 5 gigahertz band. There are some basic characteristics of radio wave propagation that affect radio communications and will be discussed next.

1. RF Propagation

In the study of RF propagation, the most basic model of radio wave propagation, where the signal is generated by an antenna and propagates evenly in all directions with no interference or obstacle to the destination antenna, is known as the free space model. The main characteristic that affects signals in this free space model is fading or path loss. Basically, in this model, the farther away from the transmitting antenna, the weaker the

signal strength becomes. In fact, the signal strength is approximately inversely square in proportion to the distance away from the transmitter in this free space model [Ref. 12].

In the real world environment however, radio waves do not travel a simple direct path. When radio waves are transmitted in the real world environment they produce reflection, diffraction, and scattering components. Reflection is the component that occurs when the radio waves bounce back from a surface that is much larger than the wavelength prior to reaching the destination. Diffraction occurs when the wave travels through different materials like through a wall or when the wave path is bent when it encounters a sharp edge. Scattering occurs when the radio wave encounters rough surfaces with objects that are smaller than the wavelength of the radio signal [Ref. 13]. Figure 2.1 below shows situations that can result in reflection, diffraction, and scattering as the radio wave travels from source to destination.

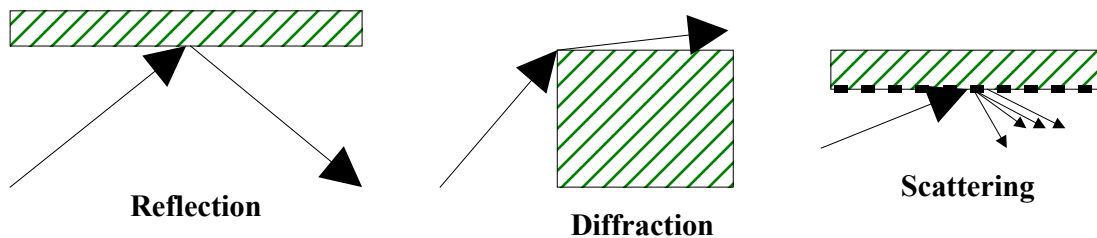


Figure 2.1 – Components of Radio Wave Propagation

2. Path Loss

The amount of fading depends on many things including the physical medium that the signal is transmitting through, the orientation of the antenna, other signals that it may interact with, and obstacles in the path of the signal. All of these elements in the environment of the signal will have an affect on the signal power. The main effect of path loss is the limit on the range of a RF WLAN. There are several methods to

overcome path loss. First, the transmitter can increase its power output to increase its range. However, government regulations, cost of equipment, and energy consumption are some of the obstacles to increasing power output. In the United States, the FCC regulates how much power and at what frequency certain equipment can transmit. This is to prevent equipment from interfering with each other. Additionally, with handheld devices that operate on batteries, the amount of time between charges could be reduced greatly if the WLAN PC card increases power consumption to increase power output. Range can also be increased by using a directional antenna. A directional antenna is one that has been designed to direct the energy of its transmission in a certain direction. These antennas are also designed to collect energy more efficiently from a certain direction and therefore are more sensitive to signals coming from that direction. Figure 2.2 shows some typical antenna shapes and their typical footprints. The first two antenna designs shown are monopole and dipole antennas, which have evenly distributed or omnidirectional footprints in the azimuth plane. These antennas will transmit and receive signal equally in all directions. A third design shown in Figure 2.2 is a parabolic type antenna. Parabolic antennas are designed to be directional, to focus the transmission and receive sensitivity in a certain direction. Directional antennas require a more precise set up because they have to be aimed. In general, they are also larger and are used in a permanent point-to-point setup.

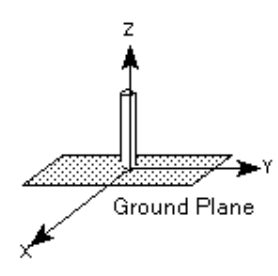
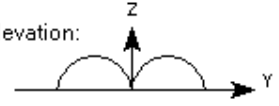
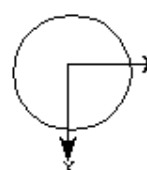
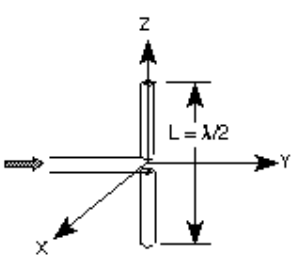
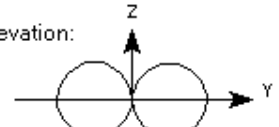
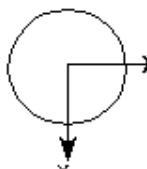
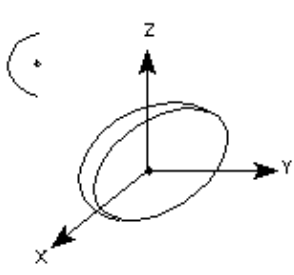
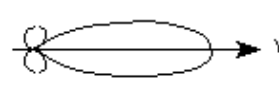
Antenna Type	Radiation Pattern	Characteristics
<p>MONOPOLE</p> 	<p>Elevation:</p>  <p>Azimuth:</p> 	<p>Polarization: Vertical as shown</p> <p>Typical Half-Power Beamwidth 45 deg X 360 deg</p> <p>Typical Gain: 2-6 dB at best</p> <p>Frequency Limit Lower: None Upper: None</p> <p>Remarks: Polarization change to horizontal if rotated to horizontal</p>
<p>$\lambda/2$ DIPOLE</p> 	<p>Elevation:</p>  <p>Azimuth:</p> 	<p>Polarization: Vertical as shown</p> <p>Typical Half-Power Beamwidth 80 deg X 360 deg</p> <p>Typical Gain: 2dB</p> <p>Frequency Limit Lower: None Upper: None</p>
<p>PARABOLIC (Prime)</p> 	<p>Elevation & Azimuth</p> 	<p>Polarization: Takes polarization of feed</p> <p>Typical Half-Power Beamwidth 1 to 10 deg</p> <p>Bandwidth: 33% or 1.4:1 limited mostly by feed</p> <p>Frequency Limit: Lower: 400 MHz Upper: 13+ GHz</p>

Figure 2.2 – Foot Pattern For Omni Directional and Directional Antenna From Ref. [14]

3. Multipath

As mentioned earlier, multipath is produced by diffraction and reflection. Multipath is used to describe the way that signals arrive to the receiving station. For each signal transmitted, multiple signals arrive at the destination via different paths. This is because some of the signals are reflected and diffracted before they arrive at the

destination while others travel a direct path (shortest route). This situation causes signals that travel via different paths to have different transit times to get from the source to the destination. Since the propagation speed of radio signals through the air is approximately the same, signals that travel by the most direct path will arrive at the destination first, followed by other components of the main signals that have been reflected and diffracted prior to reaching the destination. In urban areas there are often many buildings and metal structures for signals to reflect off. This provides many path opportunities that contribute to high multipath. This causes a significant problem since multipath distorts signals. The result of multipath is that the resulting signal strength will vary with location and frequency. This type of interference is also called small scale fading because only small bands of frequency are affected at any given location and time. Multipath is such a serious problem that it is one of the main factors that limit a system throughput.

4. Dealing With Multipath Interference

There are several methods employed to overcome the problem of multipath. Antenna diversity is a popular method. Because multipath causes the resulting signal strength to change with location, antenna diversity takes advantage of this characteristic. By using two antennas spaced at a multiple of a quarter wavelength apart, the signals that the two antennas receive are likely to have different multipath properties and therefore will be affected differently from the path traveled. The receiver will then combine these two signals (result from multipath) to generate the best signal [Ref. 15].

Wireless systems can also use equalization to help reduce the affect of multipath. This method involves sampling the received signal, then increasing the amplitude at the frequency that has been attenuated in the path and reducing the amplitude of the signal at

the frequency that has been amplified in the path. However, this method is processor intensive especially in a fast fading situation caused by the transmitter or receiver moving or the rapid changes in the environment [Ref. 16].

Another method uses multilevel coding like Differential Quadrature Phase Shift Keying (DQPSK), which uses changes in the signal's phase to represent two bits of information [Ref. 17]. By using this method the "symbol rate" for 2 Mbps bit stream can be reduced to 1Msymbol/s [Ref. 24]. One drawback of this method is that it requires a higher signal to noise ratio (s/n) to maintain the bit error rates (BERs) [Ref. 18].

Orthogonal Frequency Division Multiplexing (OFDM) tries to overcome the multipath problem by dividing the available bandwidth into many sub-bands of frequency. The adjacent sub-bands in this system are kept orthogonal from each other so that they do not interfere with each other [Ref. 27]. IEEE 802.11a and the European's HIPERLAN 2 use this type of encoding in the physical layer. We will discuss this in detail later in this chapter.

The 802.11b standard and many other current proprietary systems use Frequency Spread Spectrum (FSS) to overcome the multipath problem. FSS takes advantage of the fact that multipath only affects certain frequency significantly at certain locations. The two main types of FSS are Frequency Hopping Spread Spectrum (FHSS) and Direct Sequence Spread Spectrum (DSSS). FHSS changes the frequency that it uses to transmit data frequently so that if multipath is affecting a certain frequency it will only lose that small portion of the bit stream until it hops to a new frequency. DSSS on the other hand spreads the signal that it is transmitting over a wide band of frequency so that if multipath is affecting one of the frequencies in this band the receiver can still receive the other

components of the transmission and the signal to noise ratio is still adequate to recover the signal. In the next section this thesis will examine FSS in more detail.

B. SPREAD SPECTRUM TECHNOLOGY

Lamarr and Antheil invented Spread Spectrum (SS) in 1942 to establish an anti-jamming control for torpedo. However, SS was not used until 1962. SS is now widely used in many communication systems [Ref. 19]. There are four different types of SS: FHSS, DSSS, Pulsed FM Spread Spectrum, and Hybrid Spread Spectrum that uses a combination of DSSS and FHSS. The basic theory of SS is to spread the transmission over a wide band of frequency. In fact, the bandwidth used to transmit the signal is many times the minimum bandwidth required to transmit the data. Some military systems use up to 1 million times the minimum bandwidth to spread the signal. Because the signal is spread out in a wide band it is difficult to jam or intercept [Ref. 21]. Another benefit that SS provides in addition to anti-jamming is low power density – as the signal is spread out the power transmitted at any frequency is reduced. This lower power density makes a spread spectrum signal less likely to cause interference to other communication systems operating in the same frequency range. To other communication systems, the spread spectrum signal appears to be a series of random noises. Spread spectrum also gives the users better privacy since the random code for hopping in FH or the chip pseudo random noise code in DSSS if kept secret can keep others from intercepting the signal. IEEE 802.11b uses both FHSS and DSSS in the physical layer. Matthews and McConnell [Ref. 20 and 21] discussed FHSS, DSSS, and the IEEE 802.11b standard in detail. This thesis will concentrate more on OFDM, which has been selected for the IEEE 802.11a and HIPERLAN 2 implementation.

C. ORTHOGONAL FREQUENCY DIVISION MULTIPLEXING (OFDM)

OFDM has become one of the most popular systems for WLAN. It has been selected for use in HIPERLAN version 2, and 802.11a. There is also a proposal to use OFDM with 802.11b to achieve a data rate of 22 Mbps. As the name implies, OFDM is a form of Frequency Division Multiplexing (FDM). FDMs divide the available bandwidth into sub-carriers. There is a cost to FDM systems however. Because the sub-carriers can interfere with each other, a guard band is placed on each sub-carrier to separate and reduce their interference with each other. This makes FDM less frequency efficient. See Figure 2.3 below.

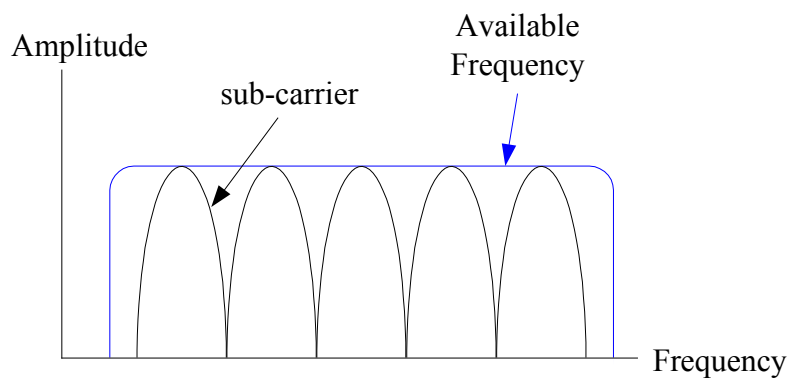


Figure 2.3 – Frequency Division Multiplexing

OFDM improves on FDM by making the sub-carriers that are adjacent to each other orthogonal. Since the sub-carriers are orthogonal they are considered to be independent of each other and a guard band between each sub-carrier is not needed. This makes OFDM more frequency efficient than FDM (See Figure 2.4 below). Some implementations of OFDM use some of the sub-carriers for error correction codes to make the transmission more reliable. This is called Coded OFDM (COFDM). In each implementation the designer then can modulate each subcarrier at a slower bit rate. With

a slower bit rate in each subcarrier, the stream of symbols that the subcarriers carry can be slowed down, the symbols are separated further from each other on the time scale (i.e. the dead space between each symbol is increased) so that the “smearing” affect that multipath has on these symbols can be ignore. In IEEE 802.11a Binary Phase Shift Keying (BPSK), Quadrature Phase Shift Keying (QPSK), and 16 Quadrature Amplitude Modulation (16QAM), and 64QAM are all possible modulation schemes depending on the required data rate. In the next section this thesis will examine how 802.11a implements OFDM in its physical layer.

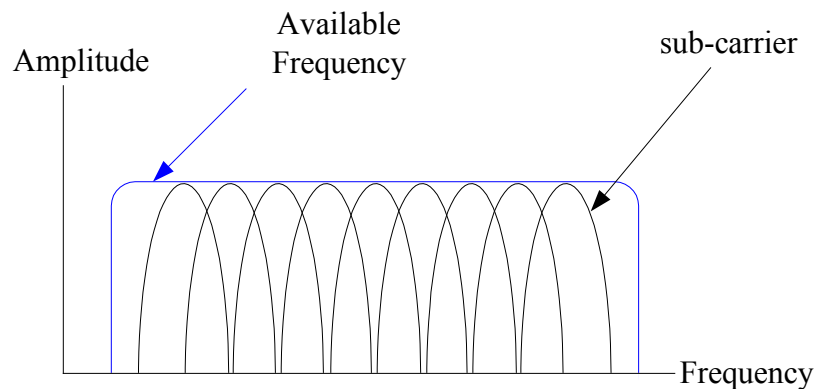


Figure 2.4 – Orthogonal Frequency Division Multiplexing (OFDM)

D. OPEN SYSTEM INTERCONNECT (OSI) LAYERS

The Open System Interconnect (OSI) model as developed by the International Standard Organization (ISO) has seven layers. The seven layers from bottom up are:

Physical (Layer 1) – The physical layer is responsible for getting the bits of information onto the communication media (air, fiber, twisted pair, etc...) to get from one point to another. It specifies the electrical and mechanical characteristic required.

Data Link (Layer 2) – The data link layer governs how the medium will be shared, for example the exponential backup rule that delays resending a packet when a

collision occurs. The data link layer specifies flow control, error control, addressing, and link management. In Ethernet, the Medium Access Control (MAC) and Address Resolution Protocol (ARP) operate in this layer.

Network (Layer 3) – Provides the means to have end-to-end communication and routing. In TCP/IP, IP resides in this layer.

Transport (Layer 4) – The transport layer performs data reliability and integrity functions. In TCP/IP, TCP resides in this layer.

Session (Layer 5) – This layer is used to maintain certain quality of service. It is also used keep a consistent connection during a login session in a multi-server load balancing system. Some implementations use this layer to keep track of a user login to recognize that the user has already been authenticated.

Presentation (Layer 6) – This layer defines the format of data to be exchanged between applications. It can also perform tasks such as encryption.

Application (Layer 7) – This is the layer where the end user application program operates (e.g., Microsoft Word and Netscape Browser).

When a computer user is using a program to communicate with another remote program, the information travels from the originator through the protocol stack from layer 7 to layer 1 of the originator computer. On the recipient end, the data travels back up from layer 1 to layer 7 to the other user. The layers in this stack are theoretical and not strict, not all applications implement the layers exactly as shown. A WLAN consists of the first two layers of this protocol stack, the physical layer and the data link layer. However, the 802.11 implementation of the data link layer called MAC layer, also

performs some of the higher layers functions to conceal that it is using a wireless medium [Ref.24]. See Figure 2.5 below for the OSI layer and 802.11a implementation.

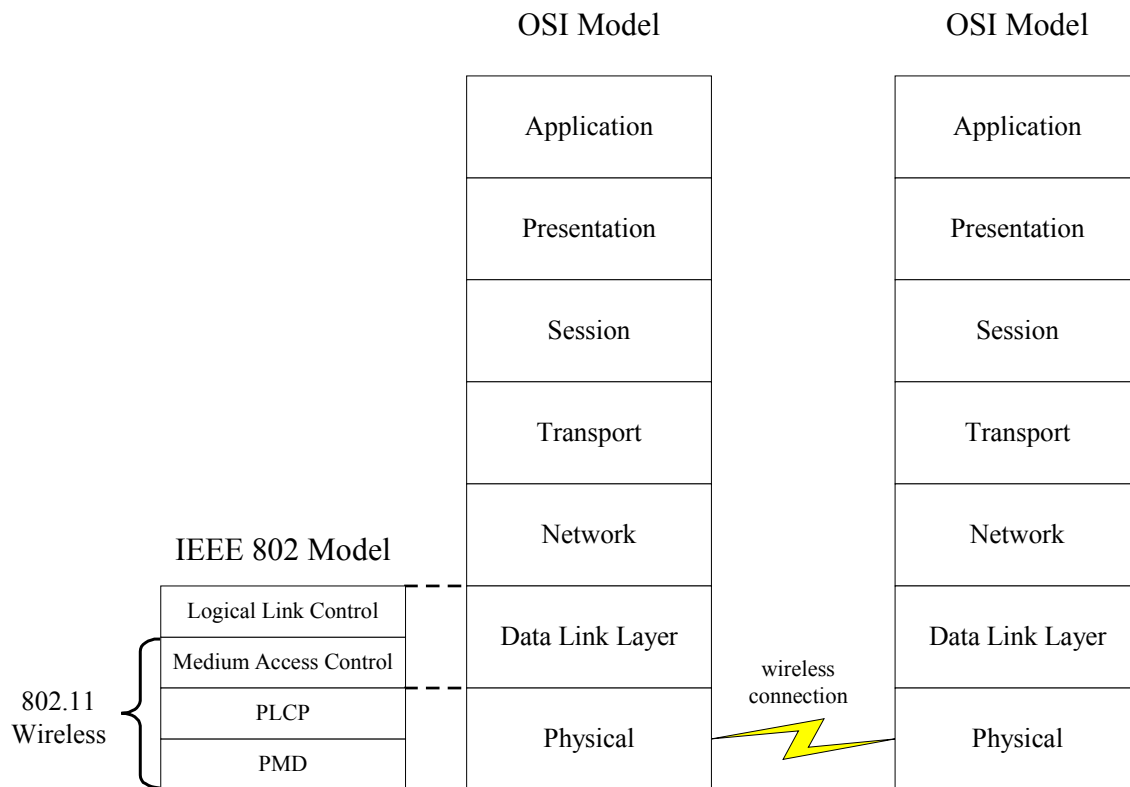


Figure 2.5 – OSI Layers and IEEE 802.11a

1. IEEE 802.11a physical layer

The 802.11a physical layer is similar to the European Telecommunications Standardization Institute (ETSI) HIPERLAN 2 physical layer. Currently, these groups are in negotiation to deploy a compatible physical layer. One standard proposed to make the physical layer of the HIPERLAN 2 standard and the IEEE 802.11b standard compatible is called the 5-GHz Unified Protocol (5-UP) by Atheros Communications. The IEEE 802.11a divides the physical layer into two sub-layers, the Physical Medium Dependent (PMD) layer and the Physical Layer Convergence Procedure (PLCP) sub-

layer. The PLCP is located above the PMD layer and just below the MAC layer. The PLCP takes the MAC Protocol Data Unit (MPDU) and converts it to a PLCP Protocol Data Unit (PPDU).

a. Physical Layer Convergence Procedure Protocol Data Unit

The PPDU is composed of the PLCP preamble, the PLCP header, the Physical sub-layer Service Data Unit (PSDU), then the tail bits, and the pad bids. The PSDU is the actual data being send. The 802.11 actually specifies five different physical mediums that can be implemented:

- Frequency Hopping Spread Spectrum at 2.4GHz
- Direct Sequence Spread Spectrum at 2.4GHz
- High Rate Direct Sequence Spread Spectrum at 2.4GHz
- Infra Red
- Orthogonal Frequency Division Multiplexing at 5GHz

The PLCP keeps the interface with the MAC layer consistent no mater which medium is being used. The PLCP translates the data from the MAC layer to the PMD sub-layer.

Figure 2.6 below shows the PPDU frame.

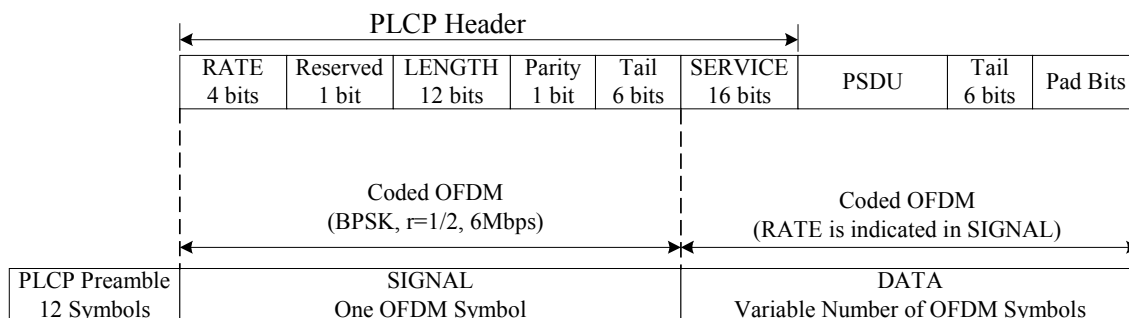


Figure 2.6 – PPDU Frame Format From Ref. [9]

- PLCP Preamble – the PLCP preamble consist of ten short symbols and two long symbols for the purpose of synchronization used to train the receiver. The total time to transmit this preamble is 16 μ s.
- RATE – the 4 bit rate field specifies one of 8 possible data rates. The available rates are: 6, 9, 12, 18, 24, 36, 48, and 54 Mbps.
- Reserve (R) – reserved for future use.
- LENGTH – A 12 bits field that specifies the number of octets in the PSDU.
- Parity (P) – used to make bit 0-16 even parity for error checking.
- Tail (SIGNAL TAIL)– 6 bits set to “0” for the tail signal.
- SERVICE – 16 bits long. Bits 0-6 are set to “0” to synchronize the receiver’s descrambler. Bits 7-15 are reserved for future use.

As indicated in Figure 2.5, the SIGNAL portion is coded with a coding rate of $R = 1/2$ using BPSK at a data rate of 6Mbps. The data portion of the PPDU frame is coded with the data rate specified in the RATE field.

b. Physical Medium Dependent (PMD)

The PMD is the last layer in the protocol stack prior to the signal being transmitted over the air. The 1999 IEEE 802.11a standard describes the PMD responsibilities in the following way:

The PMD specification establishes minimum technical requirements for interoperability, based upon established regulations at the time this standard was issued. These regulations are subject to revision, or may be superseded. Requirements that are subject to local geographic regulations are annotated within PMD specification...Operation in countries within defined regulatory domains may be subject to additional or alternative national regulations.

In other words, the PMD defines how 802.11a will work within the government regulations concerning radio transmission. The PMD takes the data frame from the PLCP sublayer. This frame is also called the PPDU frame (as shown in Figure 2.6). The PMD continues to process the data and transmits it as a radio wave. Figure 2.7

below shows the block diagram of the PMD. This diagram shows what actions have to be performed on the PPDU frame prior to transmitting it as radio waves.

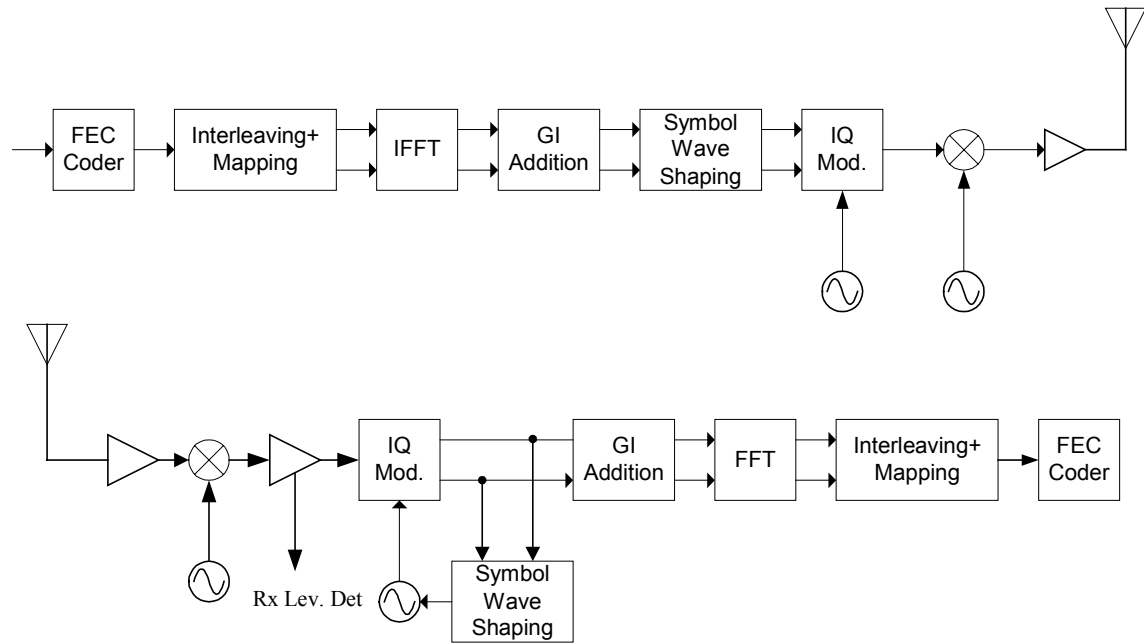


Figure 2.7 – PMD Transmitter and Receiver Functional Block Diagram After Ref. [9]

First, the PPDU frame is put through a Forward Error Correcting (FEC) Coder. For this, a convolutional encoder is used. The encoded data is then interleaved to avoid adjacent coded bits being mapped into the adjacent subcarriers and to prevent long runs of low reliability bits. The data will also be mapped into subcarrier's modulation using either BPSK, QPSK, 16-QAM, or 64-QAM. A Fast Fourier Transform (FFT) of the data will be taken and the Guard Interval (GI) will be added prior to the radio wave being generated and sent out. [Ref. 9]

c. IEEE 802.11a Frequency and Regulations

Currently the 802.11a is primarily a United States Standard. The frequencies and manner that 802.11a operates have not been approved by other regulating authorities. The 802.11a operates in the 5 GHz Unlicensed National Information

Infrastructure (U-NII) band. In CFR 47 part 15 subpart e, the FCC has allocated three 100 MHz bands for U-NII uses:

- 5.15 GHz to 5.25 GHz with up to 50 mW power transmit
- 5.25 GHz to 5.35 GHz with up to 250 mW power transmit
- 5.725 GHz to 5.825 GHz with up to 1 W power transmit

IEEE 802.11a divides the available frequency into twelve 20 MHz channels as shown in Figure 2.8 below. Each of twelve channels has 52 subcarriers by using OFDM and the subcarriers are numbered from -26 to 26 . Of the 52 subcarriers, 48 are used to carry data and 4 are pilot subcarriers used to help train the receiver for frequency offsets and phase noise. The pilot subcarriers (subcarriers -21 , -7 , 7 , and 21) are modulated using BPSK.

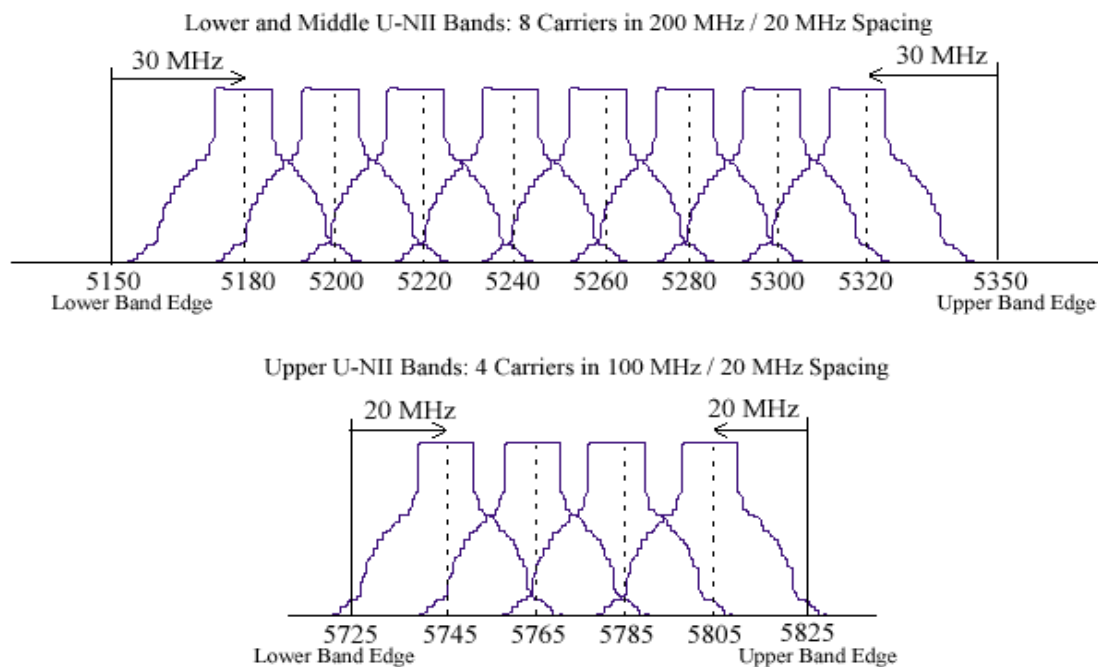


Figure 2.8 – OFDM Frequency Channels From Ref. [9]

These frequency bands give 802.11a an advantage over 802.11b at 2.4 GHz. First, the Industrial Scientific Medical (ISM) band in the 2.4 GHz only offers 83

MHz of spectrum while the U-NII band offers a 300 MHz bandwidth. Additionally, the 5 GHz range is less crowded with other equipment. 802.11b has to contend with many medical, industrial, scientific, and consumer equipment operating in this frequency. In fact cordless phones and microwave ovens are some of the common equipment that can interfere with 802.11 operations. There are limitations to using IEEE 802.11a outside the United States however. In Asia and Europe, the availability of frequencies in the 5 GHz range for WLANs is much more limited, compared to the U-NII band in the United States. In Japan, only the lower 100 MHz band in the 5 GHz range is available for WLANs and in Europe only the lower 200 MHz is available. Furthermore, currently HIPERLAN 2 is the only standard allowed to operate in Europe. IEEE 802.11a needs to implement two additional features: Dynamic Frequency Selection (DFS), and Transmit Power Control (TPC) before it can be used in Europe. DFS and TPC provide the transmitter with the capability to select a different frequency channel and lower the power output when the system detects interferences [Ref. 23].

2. 802.11a Medium Access Control (MAC) Layer

One of the advantages that IEEE 802.11a has over other wireless technologies is that it uses the same MAC protocol that IEEE 802.11b uses. This MAC protocol is familiar to many developers. Functions of the MAC Layer include:

- Control access to the wireless medium.
- Provide a reliable data delivery service.
- Perform privacy and security protection for the data

a. Control Access to Wireless Medium

Because most wireless systems cannot receive signals while they are transmitting, they cannot detect collision. Therefore the IEEE 802.11 standard does not use Collision Detection (CD) as with the wire Ethernet standard. The IEEE 802.11 standard instead uses Collision Avoidance (CA). In the IEEE 802.11 standard, if a system wants to transmit it first listens to the medium to determine if the medium is busy. If the medium is busy, the IEEE 802.11 MAC layer will perform an exponential backoff algorithm prior to attempting to transmit again. The IEEE 802.11 MAC layer also uses a Network Allocation Vector (NAV). The NAV acts as a virtual medium sensor. Even when the physical layer indicates that the medium is available NAV will indicate how long the system needs to wait prior to being able to send its data.

The MAC layer has two methods for medium access control – Distributed Coordination Function (DCF) and Point Coordination Function (PCF). DCF operates very much like the wire Ethernet standard that does not use a central controlling computer to control access to the medium. The PCF mode of operation uses a Point Coordinator (PC), which is a function that an Access Point (AP) may perform. In this mode the PC has higher control of the medium because it delays transmission by the PC Function Interframe Space (PIFS). This Interframe Space (IFS) is shorter than the DCF IFS so the PC can gain access to the medium while the DCF unit is still waiting. Once a PC gains access to the medium it can send out a special frame to set the Contention Free Period (CFP). The CFP broadcasts to other units how long they have to wait prior to attempting to gain control of the medium.

b. Provide Reliable Data Delivery Service

The IEEE 802.11 MAC layer completes several tasks to ensure that data can be delivered reliably. As the MAC layer receives data from the higher layer it may have to fragment the data into smaller size packets and construct a MAC frame for each packet for the physical layer. The MAC data frame consists of the MAC header, the body and the Frame Check Sum (FCS) (See Figure 2.9 below).

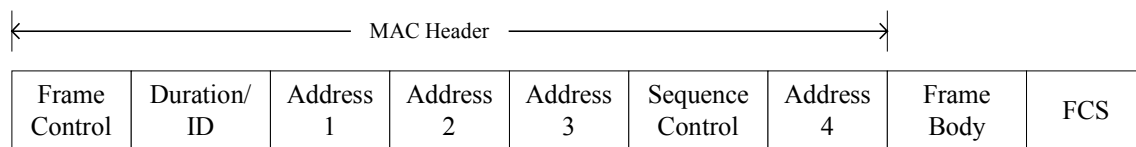


Figure 2.9 – MAC Frame after Ref. [22]

- Frame Control (2 Octets) – This field contains several parameters to specify the protocol version, frame type (management, control, or data), and frame subtype (provide additional information to the frame type). It also contains fields to specify sending and receiving Distribution System, more fragmentation, retry, power management, more data, Wired Equivalent Privacy, and order.
- Duration/ID (2 Octets) – Specifies the duration of time for the Network Allocation Vector discussed earlier or the association ID used to obtain the data buffer at the AP for a unit.
- Address 1-4 (6 Octets each) – Specify the source, destination, transmitter, and receiver address. This field also contains the Basic Service Set Identification.
- Sequence Control (2 Octets) – This field contains the fragment number and the sequence number.
- Frame Body – (0 – 2312 Octets) This field contains three items – the MSDU, the WEP's CRC called Integrity Check Value (ICV), and the WEP Initialization Vector (IV).
- FCS – (4 Octets) This field contains the 32 bit CRC of the frame body and header. The CRC is used to determine if the frame body or header have been changed during transmission.

E. WIRELESS SECURITY

Computer security is a concern that every computer network needs to address. However, the nature of WLAN operations makes this issue a much bigger concern. In a wired LAN system, one of the protections that the system has is the physical security. An attacker would need to be able to tap into the wire to access the LAN. In a wireless system, the attacker can tap into the network without actually being in the building. One attack scenario is the attacker sitting in the parking lot using a laptop computer with an 802.11 compatible card to gain access into a company LAN, while bypassing the firewall (see Figure 2.10). The developer of IEEE 802.11 realized this security risk and incorporated Wire Equivalent Privacy (WEP) into the standard.

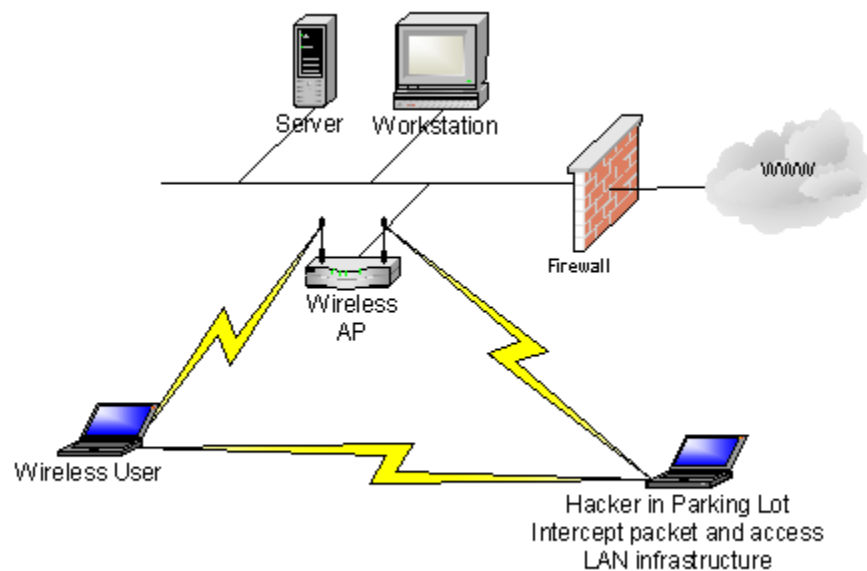


Figure 2.10 – Parking Lot Attack

1. Wire Equivalent Privacy (WEP)

WEP is a solution that IEEE 802.11 proposed to provide WLAN with a level of security and privacy that is similar to wire LAN. WEP is designed to provide three main

protections. It is supposed to protect the user from eavesdropping, unauthorized access to the network, and ensure integrity of the data. WEP uses Rivest Code # 4 (RC4) algorithm to encrypt the data for transmission. This algorithm was designed by Rivest for RSA Security. RC4 has been studied extensively with no major flaw found. RC4 is a symmetric, stream cipher. Symmetric means that it uses the same key to encrypt and decrypt the data and stream cipher means that it encrypts and decrypts one symbol at a time rather than a block of symbol at a time [Ref. 24]. In the previous section, Figure 2.7 shows the format of a MAC frame. In this frame, the frame body will be the only field encrypted in a WEP enabled installation. There have been studies that have indicated that WEP fails in all three areas that it was designed to protect. Borisov, Goldberg, and Wagner [Ref.25] have showed how an attacker can modify off the shelf equipment to provide the necessary physical layer to overcome the WEP security system. Many vendors including Cisco have claimed that they have implemented a proprietary system over the minimum 802.11 security standard to overcome the weakness that were pointed out by Borisov. The Cisco solution requires an installation of a Remote Access Dial-In User Service (RADIUS) server that performs authentication and then supplies a dynamic key for the wireless client and AP. This dynamic key changes on a per client, per session basis. This overcomes one of the major weaknesses of the WEP system [Ref. 26]. The IEEE 802.11 working group is working on improving the WEP with a new standard called the WEP2. However this new standard has not been approved [Ref. 27].

2. Standards Security Precautions

With network security it is usually a good idea to use a layered approach to provide multiple layers of security. For example a system designer may elect to put all

access points on a separate LAN, which needs to go through a firewall prior to accessing the normal LAN. In Chapter 4 of this thesis the issues of protecting an Internet Database Server will be examined.

F. WIRELESS LAN ISSUES

There are many other issues that a WLAN system has to resolve. Some of the issues are hidden node, fair contention, WLAN configuration, WLAN integration into the wire LAN, and competing for radio frequency. Mathews and McConnel address many of these issues in their theses [Ref. 20 and 21]. They also examined the issue of finding the best palm computer that can implement these WLAN systems.

THIS PAGE INTENTIONALLY LEFT BLANK

III. TESTING OF WLAN COMPONENTS

In this chapter several configurations of the WLAN will be tested. One of the tests that have not been conducted in previous theses is how well access points work as radio repeaters. This configuration is useful in situations where the access point's range is not adequate to reach all areas of coverage necessary, and it is not possible to extend the network wire to bring the access point closer to the remote user. Figure 3.1 shows an example of this situation. Access Point 1 (AP #1) does not have the necessary range to reach the remote user. In this situation, AP #2 acts as a radio repeater that can extend the reach of AP #1 to the remote user. AP #2 does not require a wired network connection to the LAN backbone, however AP #2 does require access to an electrical outlet to power itself.

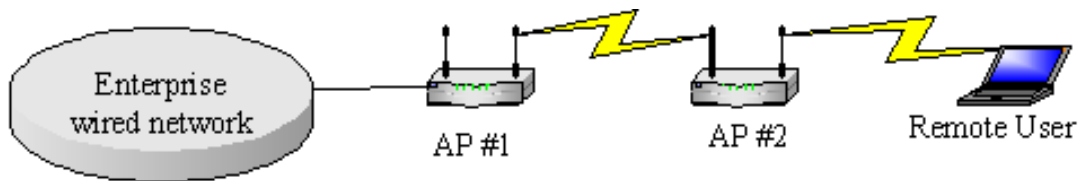


Figure 3.1 – WLAN With Access Point as Radio Repeater

A. SELECTION OF EQUIPMENT FOR TESTING

The IEEE 802.11 standard currently does not provide specifications for an access point that can operate as a radio repeater and many vendors' access points cannot operate in this mode. The WaveLAN AP by Lucent and several models of the Cisco Aironet do have the ability to operate in this mode. However the former requires additional software and hardware. Since the 802.11 standard does not specify this mode of operation, the vendor implementations provide no interoperability.

Due to the availability of equipment, only the Cisco Aironet access points were tested with the Cisco 340 and the Lucent WaveLAN Gold PC Card in this thesis. The three models of the Cisco Aironet access points that were tested are: the AP-4800E, the AP342E2C, and the AP352E2C. All of these access points are 802.11b High Rate (HR) compatible. At the remote user end, a portable computer with an 802.11b HR compatible PC card was used. The 802.11b HR compatible PC cards used were the WaveLan Gold card and the Cisco 340 PC card.

B. EQUIPMENT USED TO SUPPORT TESTING

The tests conducted required the use of supporting equipment. However, measures were taken to ensure this equipment did not influence or skew the results. For example, during testing all unnecessary programs on the remote computer and the desktop computer were shutdown and the Ethernet hub did not have any network connections other than those supporting the test. Additionally, a baseline test was conducted to ensure that that system is capable of measuring the wireless system. This test will be discussed in more detail later in this chapter.

The following is a summary of the supporting equipment used during testing:

- ❖ Desktop computer: Dell Dimension XPS R400
 - Intel Pentium II processor operating at 400 MHz
 - 128 MB RAM
 - Windows 2000 Professional Operating System
 - WS_Ping ProPack
- ❖ Remote client computer: Dell Latitude CP laptop computer
 - Intel Pentium MMX processor operating at 233 MHz
 - 96 MB RAM
 - Windows 98 Operating System

- WS_Ping ProPack
- Linksys 10/100 Integrated PC Card (used for baseline testing).



Figure 3.2 – Linksys 10/100 Integrated PC Card From Ref. [28]

- ❖ Ethernet hub: 3Com OfficeConnect Dual Speed Hub
 - Model number: 3C16750B



Figure 3.3 – 3Com OfficeConnect Dual Speed Hub 8 From Ref. [29]

C. SOFTWARE USED FOR EVALUATION

The tests conducted utilized the software utilities provided by Cisco and Lucent that is included with the purchase of their wireless hardware. Additionally, a third party utility was used to determine the bandwidth of the wireless connection.

1. Cisco Utilities

The three Cisco programs used for testing and evaluation are: Aironet Client Utility, the Link Status Meter, and the access point program. The Link Status Meter program (see Figure 3.4 below) was use to determine which access point the client computer is associated with. It also shows the signal strength and signal quality. However these values are given as a percentage and cannot be compared to other vendors' systems. The Aironet Client Utility program measures the link quality and allows the user to setup the card's configuration.

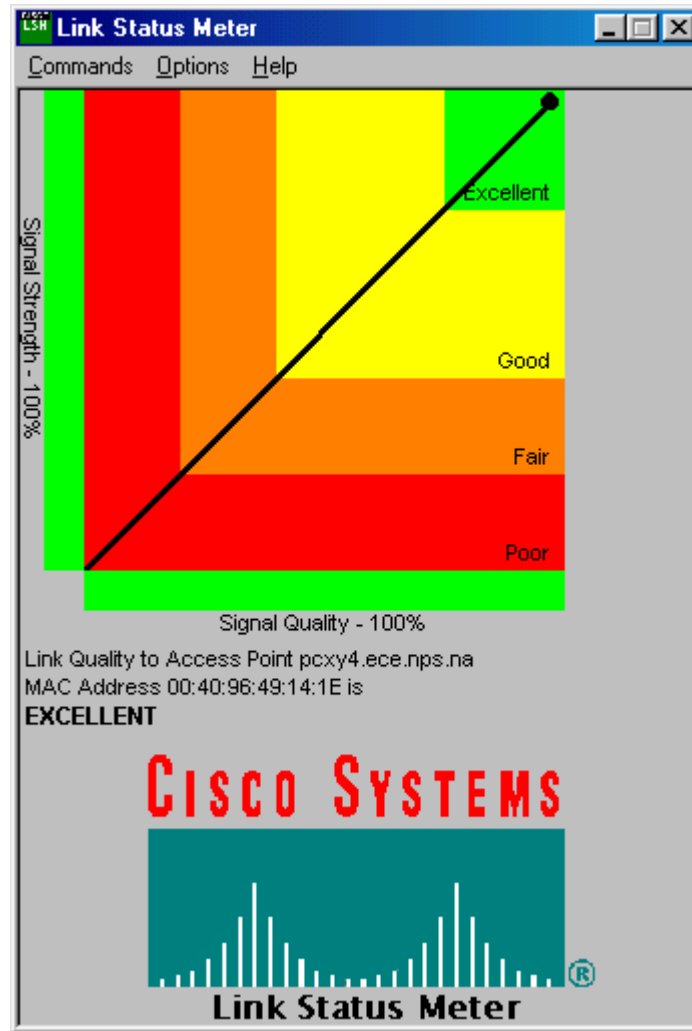


Figure 3.4 – Cisco Link Status Meter Program

The Cisco access points are accessed through a web browser and provide key status information. One of the most useful data provided by the AP, that was essential during the test, is identifying which wireless unit is associated with it. Figure 3.5 below shows the Cisco 340 AP association table, which listed three devices. The top device is a Cisco 350 AP which is a radio repeater using this the Cisco 340 AP as the root (parent node) to connect to the network. The second device listed is itself, and the third device is a laptop with a Cisco 340 PC Card adapter associated directly with the Cisco 340 AP.

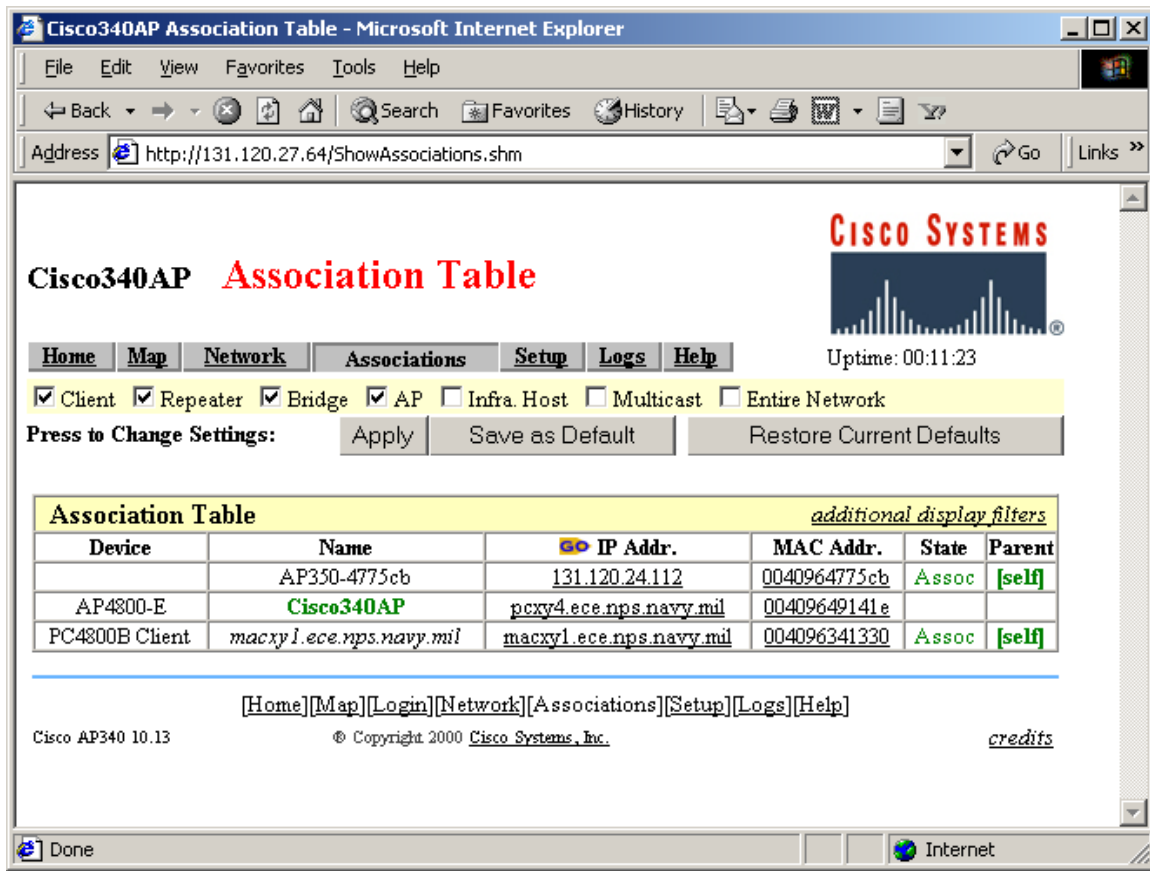


Figure 3.5 – Cisco 340 Access Point Association Table

2. Lucent WaveLAN Utilities

The WaveLAN 802.11b compatible PC Card came with a WaveMANAGER/CLIENT utility program. This program provides similar information to the Cisco Utility program. However, the WaveLAN utility program provides additional details like signal to noise ratio (SNR), signal level (in dBm), and noise level (in dBm), which are not available with the Cisco utility programs (see Figure 3.6 below).

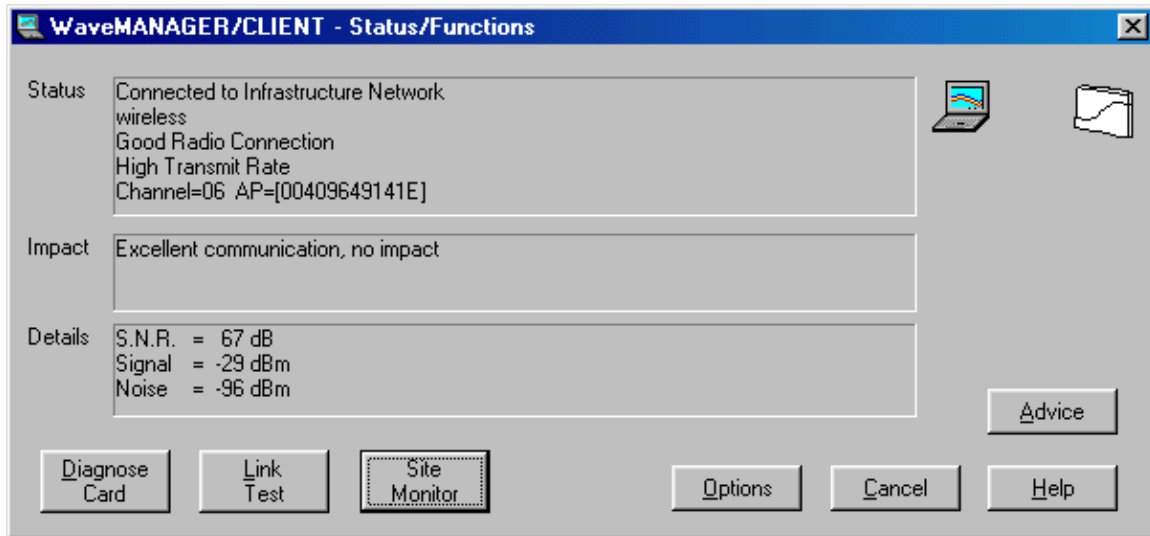


Figure 3.6 – WaveLAN’s WaveMANAGER/CLIENT Program

3. WS_Ping ProPack

In addition to the utility programs given by the wireless system vendor, a third party program was chosen to help measure the throughput of the wireless connection. The program used is a commercial program called WS_Ping ProPack by Ipswitch. This program has several utilities to help analyze a network, one of which is throughput. This program feature helps measure the throughput of a data link given an Internet Protocol (IP) address (see Figure 3.7 below).

The throughput measurement portion of the WS_Ping ProPack program allows the user to specify the following parameters:

- Packet Count – The number of packets that the program will send. The client computer (at the specified IP address) will echo these packets back, so the actual number of packets that transverse the link in both directions is double this number.
- Packet size – The size of the application packet in bytes. If the number of packets specified is greater than one, the program will increase to this size. The earlier packets in the test series will have smaller size than the size stated in this field. Additionally, if the “packet size” is greater than 1480 bytes the IP layer will fragment this into multiple smaller packets for

transmission. This is because the Maximum Transmission Unit (MTU) of the link (the MTU of the Ethernet) is 1500 bytes less the 20 bytes ICMP's header.

- Timeout (ms) – The amount of time that the program will wait for an echo reply from the remote host prior to timeout.
- Delay (ms) – The amount of time that the program will wait between packets if multiple packets were specified

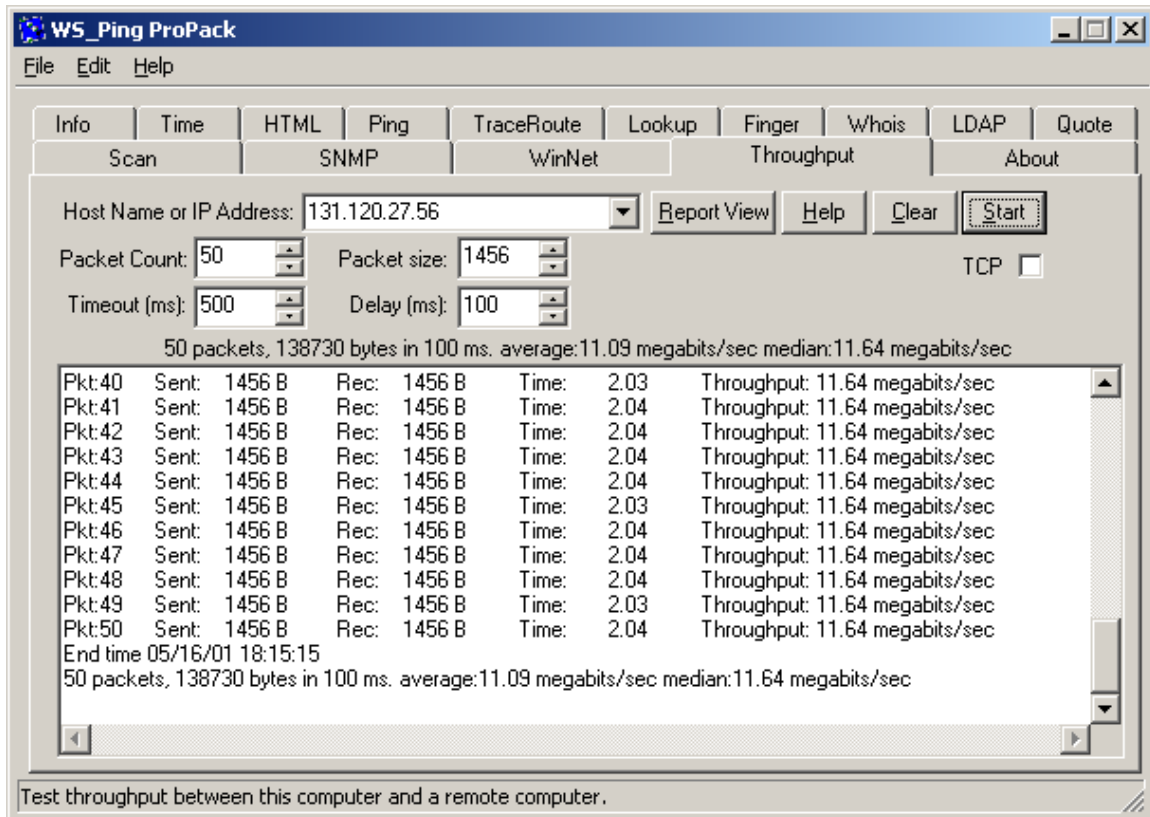


Figure 3.7 – WS_Ping ProPack Throughput Measurement

The WS_Ping ProPack software came with little documentation to explain the process it uses to measure the throughput. However, using a packet sniffer program to capture the network traffic during testing shows that the program sends out Internet Control Message Protocol (ICMP) echo request messages with the optional data field filled with data equal to length specified in the packet size field. Once it receives the

echo reply message back, it calculates the throughput by taking the amount of data transferred, which is twice the amount of the packet size multiplied by the number of packets, and divides by the amount of time it took to receive the echo replies. There is a certain amount of estimation in the calculation; the program ignores the Ethernet and ICMP overhead and any processing delay that the remote computer requires. For this testing, the following parameters will be used:

- Packet Count = 100
- Packet Size = 15,000
- Timeout (ms) = 500 and 1000
- Delay (ms) = 200

The packet size of 15,000 bytes was selected because preliminary testing shows that the throughput for the link increases when the packet size is increased. The throughput levels out around a packet size of 14,000 bytes. This could be the result of the higher layer processing delay. For example when a large size packet is specified the higher layer will only send out one large packet. The fragmentation into smaller packets takes place in the lower layers, which should be faster than the higher layer formatting a packet for transmission. However, when the throughput is measured the delays for all of the layers are included. This would cause larger packets, which require less high layer operations, to be quicker indicating a higher throughput.

D. EQUIPMENT TESTED

The equipment tested include a Cisco 340 PC Card Client Adaptor, a WaveLAN Gold PC Card Adapter, a Cisco 4800 series access point, a Cisco 340 access point, and a Cisco 350 access point.

1. Cisco 340 and Lucent WaveLAN Gold PC Card Client Adaptors

The Cisco 340 and the WaveLAN are both 802.11 b High Rate (Direct Sequence Spread Spectrum) compatible PC Cards (see Figure 3.8). The power output of the Cisco card is 30 mW and the WaveLAN card power output is approximately 32 mW (15dBm specified).



Figure 3.8 – Cisco 340 and WaveLAN Card from Ref. [30]

2. Aironet 4800, Cisco 340, and Cisco 350 Access Points

The Aironet 4800, Cisco 340, and Cisco 350 shown left to right top to bottom in Figure 3.9 were used in the testing. The 4800 is the oldest of the three models and all units are IEEE 802.11 HR compatible. The power output of the Aironet 4800 and Cisco 350 are 100 mW while the power output of the Cisco 340 is 30 mW.



Figure 3.9 – Aironet 4800, Cisco 340 and Cisco 350 Access Points

E. BASELINE TESTING

A baseline test was conducted with a Linksys 10/100 Integrated PC Card to ensure that the system is capable of communicating over the advertised 11 Mbps maximum speed specified in 802.11b High Rate (HR) wireless systems. For this baseline test a server was connected via the 3Com Ethernet hub to the Dell Latitude laptop computer with a Linksys 10/100 Integrated PC Card (see Figure 3.10 below).

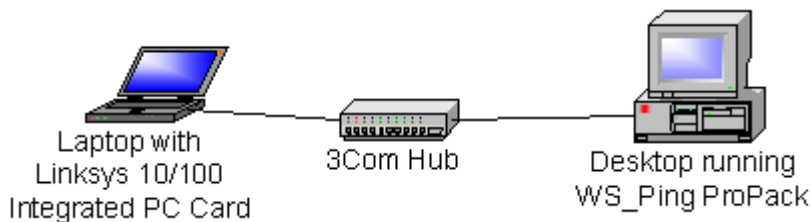


Figure 3.10 – Baseline Testing Configuration

Two sets of tests were conducted with the above configuration. In the first set the WS_Ping ProPack was running on the desktop computer, in the second set the program was running on the laptop. The results for the last twenty packets are shown below in

Table 3.1. One of the problems with the WS_Ping ProPack is that when a packet size is specified for a multiple packet test set, the program gradually increases to the specified packet size rather than test all packets at the specified size. In the test shown in Table 3.1 below the program did not reach the specified packet size until packet ninety-five. During the calculation of the throughput, the following formulas was used:

$$Throughput = (ByteSent + Byte Rec) \times \left(\frac{8bits}{byte} \right) \div Time(sec)$$

The WS_Ping program rounds the time to the nearest millisecond, which is considered to be hindered by the system clock resolution capability of most desktop computers. When analyzing the data this thesis will only use the results for ICMP packets of the specified size (15,000 bytes). For a test series of 100 packets with the specified ICMP packet size of 15,000, WS_Ping was found to have only sent six packets at the specified size.

Pkt	Sent (byte)	Rec (byte)	Desktop to laptop		Laptop to Desktop	
			Time (ms)	Throughput (Mbps)	Time (ms)	Throughput (Mbps)
81	13,024	13,024	16.07	13.02	31.06	6.72
82	13,174	13,174	16.07	13.17	17.07	12.39
83	13,324	13,324	17.00	12.54	17.06	12.54
84	13,474	13,474	17.03	12.68	17.09	12.68
85	13,624	13,624	17.04	12.82	18.00	12.11
86	13,774	13,774	17.06	12.96	18.01	12.24
87	13,924	13,924	17.08	13.10	18.01	12.37
88	14,074	14,074	17.09	13.24	33.00	6.82
89	14,224	14,224	32.08	7.11	18.08	12.64
90	14,374	14,374	32.07	7.18	19.03	12.10
91	14,524	14,524	33.00	7.04	19.01	12.23
92	14,674	14,674	18.07	13.04	19.03	12.35
93	14,824	14,824	18.08	13.17	19.06	12.48
94	14,974	14,974	33.05	7.26	19.06	12.60
95	15,000	15,000	19.01	12.63	19.07	12.63
96	15,000	15,000	18.09	13.33	19.05	12.63
97	15,000	15,000	33.05	7.27	19.06	12.63
98	15,000	15,000	18.09	13.33	19.06	12.63
99	15,000	15,000	19.01	12.63	19.08	12.63
100	15,000	15,000	19.00	12.63	19.07	12.63

Table 3.1 – Baseline Testing with Linksys 10/100 PC Card

In this baseline testing, the average throughput for ICMP packets with the size of 15,000 bytes were:

- 11.97 Mbps for desktop to laptop
- 12.63 Mbps for laptop to desktop

This baseline testing shows that the rate of throughput of the test equipment, excluding the wireless system, is sufficient to measure the throughput of the 802.11b HR compatible system. The 802.11 HR standard advertised rate is 11 Mbps, with the actual maximum throughput around 5 Mbps, as shown by McConnel [Ref. 21]. These results also infer that future testing of 802.11a, with the advertised rate of up to 54 Mbps, may be constrained by other components of the system outside of the wireless system. The low

bandwidth achieved by the 10/100 Linksys PC Card in this test could be attributed to the PC Card Bus, which is operating in 16-bit mode. To operate with higher throughput capability, a switch to a PC Card with CardBus specifications may be required. The CardBus specification is a newer standard for the PC Card, which utilizes a 32-bit bus and has a higher throughput. A series of tests were conducted with a Linksys 10/100 Integrated CardBus PC Card, which achieved an average throughput of 34.28 Mbps for the 15,000 byte ICMP packets. All of the wireless cards tested in this thesis were 16-bit bus width. The testing of packet transit time, as measured by WS_Ping, is round trip time. Therefore, the throughput for desktop to laptop and from laptop to desktop is essentially the same after disregarding anomalies and the processing difference between the laptop and desktop computer. This thesis will only test from desktop to laptop from this point forward.

F. ONE ACCESS POINT

In the next test, the throughput from a client with a Cisco 340 PC Card and the WaveLAN Gold card to a Cisco 340 access point was measured (see Figure 3.11 below). The distance from the AP and the wireless laptop is approximately thirty-five feet. The results for the test of the above configuration is given below in Table 3.2

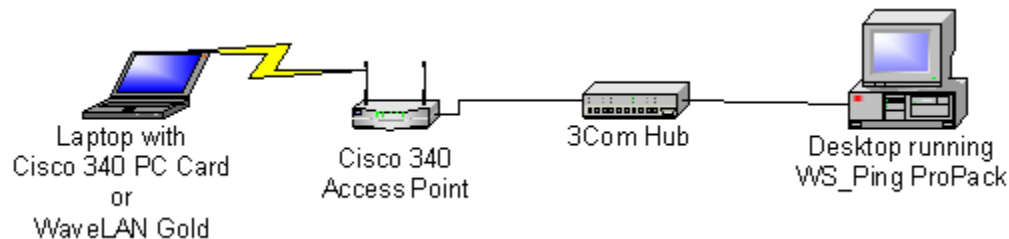


Figure 3.11 – One Access Point to Wireless Client

Pkt	Packet Size (byte)	Desktop to laptop Cisco		Laptop to Desktop WaveLan	
		Time (ms)	Throughput (Mbps)	Time (ms)	Throughput (Mbps)
95	15,000	44.01	5.45	46.00	5.21
96	15,000	43.06	5.58	42.01	5.71
97	15,000	43.05	5.58	44.00	5.45
98	15,000	44.07	5.45	44.05	5.45
99	15,000	43.06	5.58	42.03	5.71
100	15,000	43.05	5.58	45.01	5.33

Table 3.2 – One Access Point Results

The results show that there is no real difference in throughput between the Cisco PC Card and the WaveLAN PC Card. The average throughput for 15,000 byte ICMP packets was measured to be:

- Cisco card average 5.54 Mbps
- WaveLAN PC Card average 5.48 Mbps

G. ONE RADIO REPEATER

In this part, the network is setup similar to Figure 3.1 except AP#1 is connected to the 3Com hub, which is connected to the desktop instead of AP#1 connecting to the LAN backbone (see Figure 3.12 below). The wireless cards' utilities programs and the APs' association tables were used to ensure that the laptop is accessing the network through the radio repeater (Rpt #1) and not AP#1. AP#1 is the Cisco 340 series AP and Repeater 1 (Rpt #1) is the Cisco 350 series AP. Both the Cisco and the WaveLAN cards were used with the laptop computer for this testing. In this testing, Rpt #1 was placed approximately thirty-five feet from AP #1, and the laptop client was placed approximately an additional thirty-five feet away from Rpt #1. The WS_Ping program timeout parameter was increased to 1000 ms. Results from the test are shown below in Table 3.3.

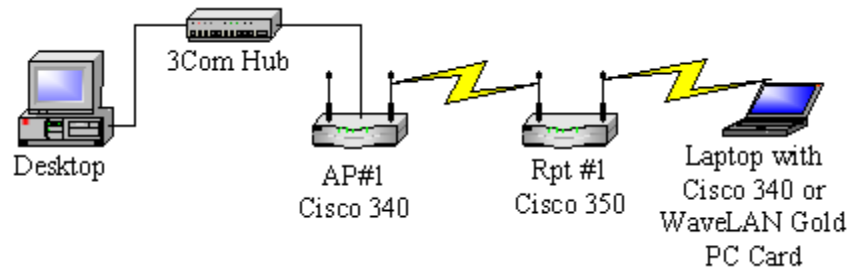


Figure 3.12 – One Repeater Wireless Network

Pkt	Packet Size (byte)	Desktop to laptop Cisco		Laptop to Desktop WaveLan	
		Time (ms)	Throughput (Mbps)	Time (ms)	Throughput (Mbps)
95	15,000	74.07	3.24	90.04	2.66
96	15,000	74.04	3.24	84.05	2.85
97	15,000	75.06	3.20	85.06	2.82
98	15,000	76.07	3.15	87.08	2.75
99	15,000	73.08	3.28	99.00	2.42
100	15,000	76.01	3.15	82.08	2.92

Table 3.3 – One Radio Repeater Accessing Through Repeater

This test shows that the difference in throughput between the Cisco PC Card and the WaveLAN PC Card is minimal. The average throughputs for the 15,000-byte ICMP packets were:

- 3.21 Mbps for the Cisco card
- 2.73 Mbps for the WaveLAN card

A second test was conducted utilizing the above configuration with the client accessing the network through AP #1, while the radio repeater is still associated with AP #1. This test was conducted to determine if the throughput to an AP is decreased when that AP is supporting a repeater even when the repeater does not have any client associated with it. For this test the repeater was placed approximately thirty-five feet from the access point and the laptop was also about thirty-five feet from the access point in a different direction. The results for this test are shown below in Table 3.4.

Pkt	Packet Size (byte)	Desktop to laptop Cisco		Laptop to Desktop WaveLan	
		Time (ms)	Throughput (Mbps)	Time (ms)	Throughput (Mbps)
95	15,000	45.03	5.33	44.01	5.45
96	15,000	45.02	5.33	43.01	5.58
97	15,000	48.03	5.00	41.08	5.85
98	15,000	47.00	5.10	45.02	5.33
99	15,000	47.00	5.10	43.03	5.58
100	15,000	46.09	5.21	42.00	5.71

Table 3.4 – One Radio Repeater Accessing Through Access Point

The resulting average throughputs were:

- 5.18 Mbps for the Cisco PC Card
- 5.58 Mbps for the WaveLAN PC Card

This is approximately the same as the throughput for the case of the access point that is not associated with any repeater. In fact, the average throughput for the WaveLAN adapter was higher for this set of tests as compared to the one access point no repeater test discussed earlier.

H. TWO RADIO REPEATERS

An experiment was conducted to determine the throughput of the system when the link consists of 2 radio repeaters. In this configuration the connection was from the desktop computer to the hub to AP #1 to Rpt #1, which was approximately thirty feet from AP #1. Rpt #1 relays the data to the second repeater (RP #2), which was placed thirty-five feet from Rpt #1 in the opposite direction of AP #1. Rpt #2 relays the data to the remote client, which was placed thirty-five feet from Rpt #2 (see Figure 3.13 below). The results given in Table 3.5 show that there is a significant decrease in throughput when compared with other configurations tested. Also note that packet ninety-six of the WaveLAN test resulted in a timeout. This means that one second (1000ms) after

WS_Ping sent an echo request message to the remote client, it still did not receive the echo response back and the packet is considered lost.

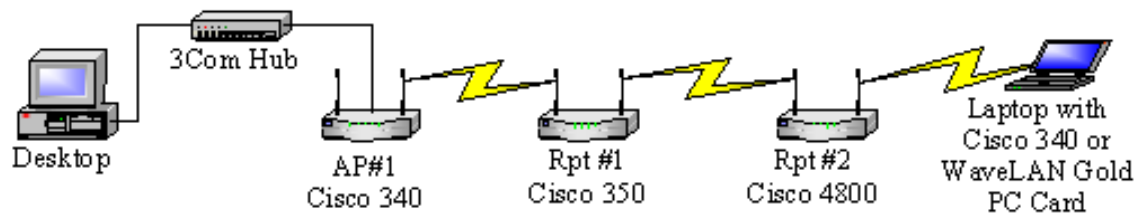


Figure 3.13 – Two Repeaters Wireless Network

Pkt	Packet Size (byte)	Desktop to laptop Cisco		Laptop to Desktop WaveLan	
		Time (ms)	Throughput (Kbps)	Time (ms)	Throughput (Kbps)
95	15,000	260.07	923.07	326.01	736.19
96	15,000	265.09	905.66	timeout	
97	15,000	254.04	944.88	329.05	729.48
98	15,000	280.00	857.14	333.03	720.72
99	15,000	269.01	892.19	326.04	736.19
100	15,000	291.02	824.74	288.07	833.33

Table 3.5 – Two Repeaters Accessing Through Second Repeater

The average throughput for a client accessing the network through a second repeater was:

- 891 Kbps for the Cisco PC Card
- 751 Kbps for the WaveLAN Card

One difference with this series of test results when comparing with previous tests is that there was a peak in the throughput with ICMP packet size between 3,000 to 5,000 bytes (see Figure 3.14 below). The average throughput for an ICMP packet of this size was 1.48 Mbps, almost twice as high as for packets of 15,000 bytes. This suggests that the network capacity had reached a maximum at this point and larger packets, which caused more fragmentation resulting in many more packets, overwhelmed the system and

caused the throughput to drop. The same plot for the WaveLAN PC Card did not show a similar peak.

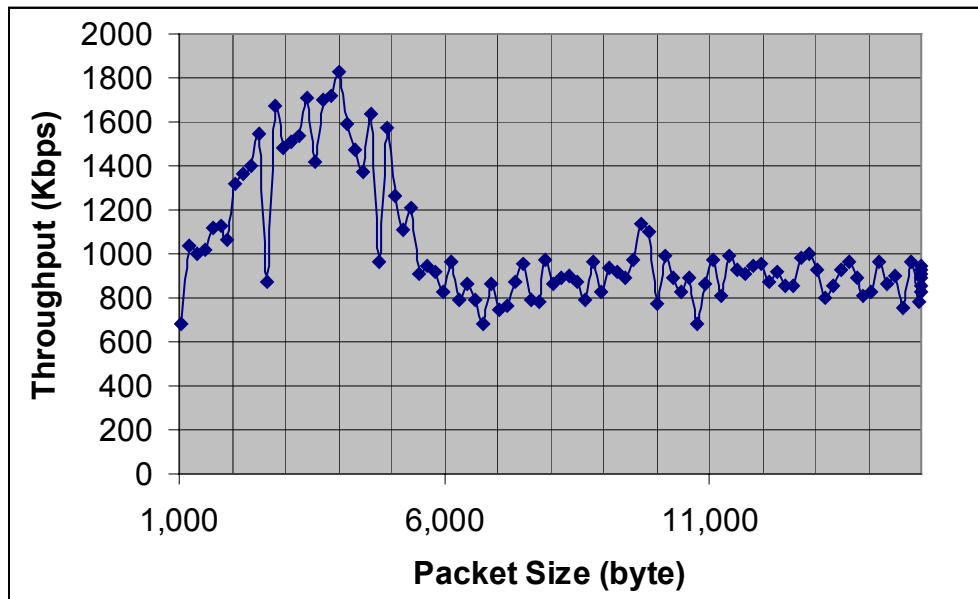


Figure 3.14 – Two Repeaters Network Results for Cisco PC Card

I. DISCUSSION OF RESULTS

These tests show that the performance differences between the Cisco and the WaveLAN cards are negligible. There was a significant reduction in bandwidth when a link had to go through two radio repeaters. The average reduction in throughput for a two repeaters network was eighty five percent. This amount of throughput reduction seems logical because when an access point acts as a repeater half of the time available is devoted to retransmission of incoming packets. When a link consists of one radio repeater the reduction in throughput was almost fifty percent on the average. One note of the different characteristics in the two cards is that the Cisco card tends to change its association back with the root AP quicker than the WaveLAN card. There are benefits and drawbacks with this approach. If it is successful in establishing this association with

the root access point, it will have a higher throughput. However, since it tries to change this association more quickly, it sometimes loses this association and has to go back to the old association with the repeater, causing a flapping effect in critical areas located between the access point and radio repeater.

THIS PAGE INTENTIONALLY LEFT BLANK

IV. WLAN APPLICATIONS

There are currently several projects underway to implement WLAN-based applications to improve shipboard operations. One of the projects jointly under development by the Naval Warfare Assessment Station (NWAS) and the Naval Postgraduate School (NPS) is a shipboard gage calibration system. Currently shipboard gage calibrations are carried out manually and calibration data are logged using paper. Manual calibration is time consuming and prone to human errors. The objective of the joint project is to automate a major portion of the calibration process by using WLANs and wearable PCs, and thus reduce calibration time. The system under development is called the General Purpose Test Equipment (GPTE) Semi-Automated High Pressure Calibrator (GSAHPC).

A. GPTE SEMI-AUTOMATED HIGH PRESSURE CALIBRATOR

The GSAHPC system allows a user to easily obtain a gage calibration reading and capture the data into a database that can be analyzed. The GSAHPC system consists of a portable computer equipped with a memory button pen reader, a hand pump, and a pressure calibrator (see Figure 4.1 below). To use this system the user first logs in and then places the pen on the button attached to a gage to obtain the gage's ID and other information. The program on the portable computer then will walk the user through the tasks necessary to take the reading from the gage. When the calibration is completed, the data is captured in the portable computer ready for further processing.



Figure 4.1 – GSAHPC System From Ref. [31]

B. WIRELESS LAN AND THE GSAHPC SYSTEM

Wireless LAN is a natural technology to provide connectivity to the portable computer. This connectivity will give the user additional functionalities without being encumbered by a network wire. With WLAN the data collected by technicians can be uploaded to a server instantly and remotely allowing the off-ship maintenance management team to monitor the data and to order parts and services for the ship while it is still underway. This could help reduce the workload of shipboard personnel. Additionally, this is a good platform for the next goal, which is to integrate Bluetooth or some other WLAN technology with gages to allow the automated collection of data. The wireless hardware used with the portable computer could be any commercial off the shelf (COTS) 802.11 compatible PC Card, like the one tested in chapter III of this thesis. The main focus of this chapter will be on the software that is needed to make this system operational.

C. INTERNET AND DATABASES

It was determined that using a database accessible through the Internet/Intranet would provide easy connectivity for both the portable computer collecting the data and the manager viewing the data. It was determined that Java would be the best platform to meet the need of this task. Java can be platform independent through the Java Virtual Machine (JVM) and it has a wide selection of built in tools like network connectivity and Java Database Connectivity (JDBC). JDBC is an Application Programming Interface (API) that allows Java programs to access many different tabular data sources [Ref. 33].

On the database server side of the system, it was decided to use Microsoft Access and Open Database Connectivity (ODBC). ODBC is a tool that allows a single application to access different database management systems (DBMS). The application would call the functions in the ODBC, which then implement database specific calls. ODBC is a Microsoft solution to give one set of calls for different types of databases running on Windows. Both of these tools are available with the Microsoft Office Suite and the Windows operating system. Even though Access and ODBC were selected for the database server side of the system it is desirable that the system not be locked in with a particular database and operating system. This freedom will allow the server side to operate with any database and any operating system in the future. To allow for this flexibility, a commercial driver was selected that will provide the connectivity between JDBC and ODBC. This driver will allow the development to be completed quickly because it provides the JDBC to ODBC connection in the application development. The development does not have to be concerned with knowing the intricacy of ODBC (see Figure 4.2 below). Internet Database Server (IDS) made by IDS Software was selected.

In addition to providing the connectivity between JDBC and ODBC, IDS has the ability to connect JDBC with other DBMS like Oracle and Sybase via their own native interfaces [Ref. 32]. With IDS Server users can use their web browser to perform database queries and submit updates to the database. IDS Server would accept the queries and generate the HTML response back to the user browser.

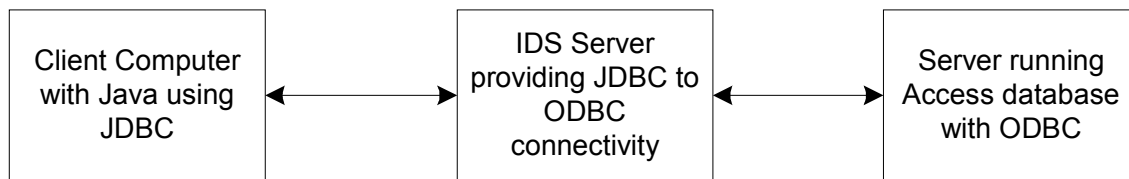


Figure 4.2 – Connectivity Between JDBC and Database

1. IDS Configuration

IDS Server has several configuration options. IDS Server can work with a web server or it can act as a web server itself (see Figure 4.3 below). In the test configuration the Database Server (Microsoft Access), IDS Server, and the web server (Microsoft IIS) were installed on the same computer. In other words, programs that are shown in Figure 4.3 as being installed on Computer A and Computer B were installed on the same computer. IDS described this configuration as running IDS Server inside a web server. This configuration was selected because Microsoft IIS provides better performance and additional features like a secure web server. IDS warned that running the IDS Server inside a web server will degrade the performance of the web server and if the IDS Server crashes it will cause the web server to crash also. This is not a concern for our system since the only purpose of the web server is to support the IDS Server. IDS offers many methods to access the database including: HTML Extensions, Java applications, and Java applets. All three of these methods were tested.

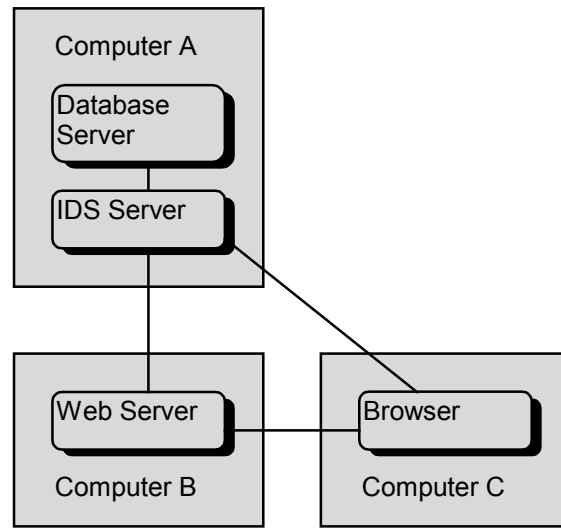


Figure 4.3 – IDS Configuration From Ref. [32]

2. Submission of Data

During the development, the original plan was for the calibration data collected to be submitted directly to the web server through Java applets as it is being collected. This method would allow a no installation solution. This is because all of the programs required to collect and submit calibration data could be installed on the IDS/web server. When the portable computer needs the application to gather and submit calibration data it would download the Java applets from the server and runs them inside the web browser. The only program that is required to run on the portable computer is a web browser. This solution was not selected by NWAS because they desired that the system be capable of working when the network or the server is down so that the technician can continue to collect calibration data and submit it latter. To achieve this requirement it was determined that a Java application had to be installed on the portable computer. This application would submit the data file already collected by a different program installed

on the portable computer to the database server. On the server computer all of the programs could be the same as the no installation solution.

3. IDS Server Configuration

One of the configurations tested was for the IDS Server to act as both the web server and the database server. The instructions for this installation given by IDS were straightforward. The basic steps to run the IDS Server include:

- Install the IDS Server, and make any necessary changes to the idss.ini file. Settings for some of the parameters in this file will be discussed shortly
- Configure the System Source Open Database Connectivity (ODBC), System DNS in Windows Administrative Tools to the name and location of the Database file. In the current configuration this file is Calibration.mdb.
- Start the IDS Server. This can be done through Windows, Administrative Tools, Services.

Any changes that need to be made to the IDS Server configuration are done through the “idss.ini” file. Two of the settings that may need to be changed are port and TunnelKeepAlive. The port setting specifies the port that the server will be active on, usually 80 or 12. The TunnelKeepAlive parameter is used to specify that the client-server connection will be through a HTTP tunnel. This setting is necessary when the firewall rule does not allow a persistent connection. With this parameter set, the data transfer can be fragmented and sent as HTTP contents. [Ref. 32]

4. Running IDS Inside Microsoft’s Internet Information Services

The IDS Server program can be run inside a web server like Microsoft’s Internet Information Service (IIS). This configuration provides the server with additional security like supporting Secure Sockets Layer (SSL). The following is a summary of the steps given in the IDS Server User Guide [Ref. 32] to run the IDS Server inside the IIS:

- Install the IDS Server. However, it does not have to be started.
- Copy the idss.dll file from the IDSServer\cgi directory into InetPub\scripts directory.
- Start the Microsoft IIS.
- When establishing connection to the Database use the following format on the client side (note the boldface part of the code differentiates this method versus running the IDS Server without IIS):
 1. <form action="http://web_address/**scripts/idss.dll**/query?htx=file://staff.htx" ...>
 2. con=drv.connect("jdbc:ids://web_address/**scripts/idss.dll**/conn?dsn=mydb", null);

5. Client Java Application

A file submission program was written in Java to submit data already collected by the technician and stored in a text file. The program reads this data file from disk and submits it to the database. The procedure to establish the connection to the database using IDS Driver is similar to using a java.net class. Section 4.4 of IDS Server User Manual [Ref. 32] discusses the format of the (Universal Resource Locator) URL for the IDS driver. If the client-server connection is through a firewall that does not allow a persistent connection, setting the "http=1" will cause the connection to use a HTTP tunnel. Many of the parameters set in the URL mirror the parameters in the idss.ini file.

A screen capture of the Java submission application is shown below in Figure 4.4. To use this application the user would type in the IP address and port number of the computer running either IDS Server or a web server. The user can use the "Browse" button to select the file to upload to the database. Finally the user clicks on the "SUBMIT" button. The log window will notify the user if the submission was successful. On the server side the IDS Server immediately updates the database. The code for this application is included in Appendix A

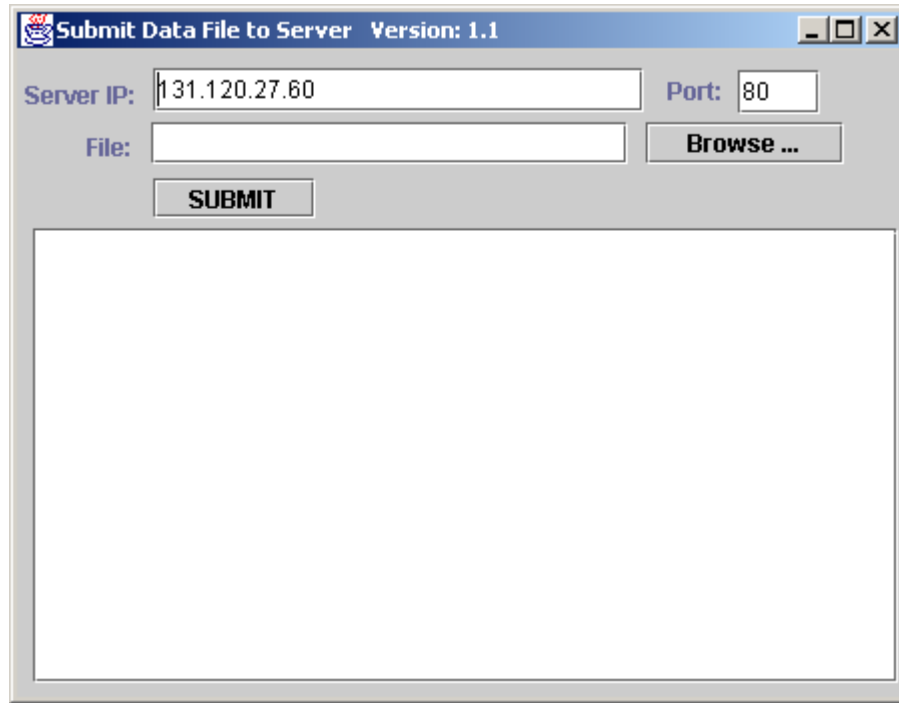


Figure 4.4 – File Submission Java Application

Currently the Java application implements minimal security protocol. One of the reasons why there is minimal security for the Java Application method is because to implement a database security IDS requires a full featured database like Oracle, or Microsoft SQL. IDS does not support database security with Microsoft Access, which is currently being used.

6. Alternatives to Java Applications on the Client Machine

Several alternatives to the Java Application running on the Client machine were examined. These alternatives are:

- Using Java Applets
- Using HTLM extensions (HTX)
- Using JavaServer Pages (JSP)
- Using Active Server Pages (ASP)

a. Using Java Applets

Java applets are Java code that have been designed to run inside an applet viewer. For most situations the web browser acts as the applet viewer. With this method, the client computer downloads applets from the server as needed and runs them. To do this, a HTML file is placed in the server like any static HTML file. This HTML file draws the outline of the page and calls the applet, which when executed will fill in the remaining data on that page. The HTML code that calls the applet to view data is included in Appendix B and the java applet that the HTML code calls is included in Appendix C. In the HTML code the following line starts the Java applet:

```
<p><applet codebase="classes" code="calibration.ViewData.class"
```

One of the difficulties with Java applets is security. In general it is unsafe to run a program that has been downloaded from the Internet. The web browser and Java have many security restrictions with applets. For example, there are methods in place to verify the source of the applet prior to executing it. Running Java applets increases security risks on the client computers. For most web browsers, applets are run inside a “sandbox” which isolates the applets by keeping them from accessing other areas on the client computer. This sandbox also restricts the ability of applets.

b. HTML Extensions

HTML extensions are commands that can be attached to HTML requests that only IDS Server will recognize. Once IDS Server recognized these commands it will perform the tasks requested. This method is simple to use; however, it is less flexible than some of the other methods since only commands that IDS implemented will work.

Additionally, these extensions are proprietary to IDS Server. A sample code to view data using HTML extensions is attached in Appendix D.

*c. **JavaServer Pages (JSP)***

JavaServer Pages is a technology developed by Sun Microsystems. Sun describes JSP as using “XML-like tags and scriptlets written in Java programming language to encapsulate the logic that generate the content for the page.” [Ref. 34] One of the main differences between JSP and Java applets is JSPs are executed on the server while applets are executed on the client computer. This eliminates the security threat of running a program from the Internet that applets have to deal with. To enable this functionality, the server needs to run a JSP container in addition to the HTTP server. When a JSP request is sent to the web server it will have a special file extension in the URL, usually .jsp, and the web server passes this request to the JSP container. The JSP container parses the JSP file into java code and static HTML. The container compiles and executes the code and combines it with the static HTML content forming a servlet. A servlet is a java code program that generates a HTML page by printing out a HTML text stream. Once the servlet is formed, whenever the JSP is called the servlet is executed and the results are sent as the response. [Ref. 35]

*d. **Active Server Page (ASP)***

ASP is a Microsoft technology that is similar to JSP. ASP supports many script languages including PerlScript, Jscript, and VBScript. The VBScript can call on Microsoft ActiveX components, which are precompiled code that offer additional functionalities. One of the disadvantages of the ASP is that the ActiveX components are currently only used by Microsoft. To take advantage of the power of ASP, it needs to be

used with a Microsoft system. [Ref.35] One of the biggest advantages of ASP is that Microsoft's IIS comes equipped with ASP. ASP codes were written to view database data and upload files to the database. To upload the file to the server computer a third party component from AspUpload was used. Even though it is possible to upload files without using this third party component, this product implements many features that are helpful and would take time to develop [Ref. 36]. The code developed during this thesis makes function calls to the AspUpload components. This code is included in Appendices E and F. The ASP code to upload files included in Appendix F, calls on the SubmitFileScript.asp. The SubmitFileScript.asp is included in Appendix G.

7. Discussion of Alternative Selection

Currently all methods discussed above for submitting and viewing gage calibration data are effective. Because of security risks the JSP and the ASP offer better service. With these two methods, the transaction could take place through a Secure Socket Layer (SSL) link between the web browser and web server. Since SSL is widely supported in most current web browsers and web servers, it could be implemented quickly. The selection between the ASP and JSP depends on plans for future deployment and market trends.

THIS PAGE INTENTIONALLY LEFT BLANK

V. CONCLUSIONS

Due to shortage of personnel the Navy is trying to improve efficiency and better utilize personnel resources. One of the areas that the Navy has studied is to use wireless LAN technology as one of the tools to achieve this goal. The objective of this thesis is to continue the work of previous theses in this area to further understand wireless LAN technology and determine how it could be applied in the area of gage calibration, one of the Navy current routine chores. Three specific tasks were completed in this thesis:

- Study technologies and standards used in WLAN, particularly the IEEE 802.11a standard.
- Test and evaluate WLAN configurations that may be used in a deployment aboard ships.
- Develop a prototype application that could be used to help collect gage calibration data.

A. TECHNOLOGIES AND STANDARDS USED IN WLAN

In this thesis, the IEEE 802.11a standard was examined. The physical layer of this standard is very different from the IEEE 802.11b standard, which is in wide use at this time. The IEEE 802.11a standard uses OFDM instead of spread spectrum used by the 802.11b standard. Furthermore, the 802.11a standard operates in the 5GHz range and offers a throughput of up to 54 Mbps in the physical layer. This compares with the 802.11b standard, which operates in the 2.4 GHz range and offers a throughput of only 11 Mbps. Currently, 802.11a standard products are not yet available. Several companies are estimating that they will have 802.11a products in 2002.

The European's HIPERLAN2 standard was also studied. This standard has many similarities with the IEEE 802.11a including the use of OFDM and the 5GHz frequency range. They also have many differences like connection-oriented link of HIPERLAN2

versus connectionless link used by the IEEE 802.11 standards. There are efforts being developed to unify these standards so that they are compatible.

B. TESTING OF WLAN CONFIGURATIONS

The ability to use the wireless access points as radio repeaters to extend the wireless LAN coverage was tested. Tests were conducted for wireless links that consist of none, one, and two radio repeaters. For the no repeater link an average bandwidth of 5.51 Mbps was measured for connecting directly to the access point. For the one and two radio repeater links an average bandwidth of 2.97 Mbps and 821 Kbps were measured respectively. Even though the bandwidth for a two repeater link was about 85 percent less than a no repeater network, it is still sufficient to support many applications like the gage calibration application.

C. DEVELOPMENT OF A WLAN APPLICATION TO COLLECT GAGE CALIBRATION

Several methods of making gage calibration data accessible through the Internet/intranet were examined and implemented. These methods include: Java applications, Java applets, and Active Server Pages (ASP). All of these alternatives are shown to be able to successfully submit gage calibration data to a database and can view data from the database through the network as expected. The source code for these implementations are included in the Appendices of this thesis. Among these methods, the ASP method is the most robust when considering security and functionalities. A fourth method that uses JavaServer Page (JSP) was examined but was not implemented. JSP offers many features that are similar to the ASP but are not limited to Microsoft systems. This could be a subject for future research.

D. RECOMMENDATIONS FOR CONTINUED RESEARCH

All of the technologies studied in this thesis are industry driven. This makes it important that constant research and evaluation are conducted to keep up with rapid changes in the industry. Because of the commercial usage of these technologies any weakness or security flaws discovered may be widely publicized. Researchers are needed to ensure that Navy keeps up with industry knowledge.

1. Future WLAN Studies

In the area of WLAN, products using the 802.11a and HIPERLAN2 standards are expected to be released in 2002. These products could double the bandwidth of WLAN and could make the old technology obsolete. A thorough examination of these products should be conducted to determine if the Navy should select these products for deployment. Evaluation of these products should include independent testing to determine the product performance. Additionally, tests should be conducted to determine if these products meet the Navy and DOD standards for shipboard deployment. Prior to deploying a WLAN system on a ship, each class of ship has to be surveyed and a plan established on where and how the access points should be installed. Even after a system has been deployed, constant research needs to be conducted to determine if there may be flaws in the implementation.

The Bluetooth standard products are starting to be released by vendors. Future studies should examine how Bluetooth can interoperate with the 802.11 standards. A Bluetooth gage could be developed to send gage readings via the wireless network directly to a database and eliminate the need for a technician to make rounds and collect gage readings.

2. Future WLAN Applications

Much work remains in this area. The Navy has formed a new Measure 21 standard, which provides many specifications on the Navy's Metrology Calibration Information System [Ref. 37]. Future studies should determine how to develop a system to be compliant with this standard. Additional work needs to be done to incorporate a program to submit files to the database with the correct format so that it could be used by other Navy's applications.

APPENDIX A – JAVA APPLICATION CODE TO SUBMIT DATA

```
//Title:      SubmitB
//Version:    1.2      24 MAY 2001
//Copyright:  Copyright (c)
//Author:
//Company:
//Description:

//Title:      SubmitB
//Version:    1.1
//Copyright:  Copyright (c)
//Author:
//Company:
//Description:

import java.lang.*;
import java.io.*;
import java.awt.*;
import java.awt.event.*;
import java.applet.*;
import javax.swing.*;
import javax.swing.event.*;
import java.sql.*;
import java.util.Properties;

import java.net.*;

public class Applet1 extends JApplet {
    boolean isStandalone = false;

    static String signature = "jdbc:ids://";
    static String conn_etc =
"/scripts/ids.dll/conn?uid='web'&pwd='opentest'&dsn='Calibration'&uid='&ssl=1&http=1";
    // static String conn_etc = "/conn?dsn=Calibration&ssl=1&http=1";
    String oldIP = "";

    JPanel jPanel1 = new JPanel();
    JLabel jLabel1 = new JLabel();
    JTextField jTextField1 = new JTextField();
    JLabel jLabel2 = new JLabel();
    JButton browseButton = new JButton();
    JTextField jTextField2 = new JTextField();
    JButton submitButton = new JButton();
    //Construct file chooser
    final JFileChooser fc = new JFileChooser();
    JScrollPane jScrollPane1 = new JScrollPane();
    JTextArea log = new JTextArea();

    //Stuff for file connection
    FileInputStream fileIS;
    FileReader fileR;
    StreamTokenizer st;
    JLabel portLabel = new JLabel();
    JTextField jTextFieldport = new JTextField();

    //Get a parameter value
    public String getParameter(String key, String def) {
        return isStandalone ? System.getProperty(key, def) :
            (getParameter(key) != null ? getParameter(key) : def);
    }

    //Construct the applet
    public Applet1() {
```

```

}

//Initialize the applet
public void init() {
    try {
        jbInit();
    }
    catch(Exception e) {
        e.printStackTrace();
    }
}

//Component initialization
private void jbInit() throws Exception {
    this.setSize(new Dimension(471, 379));
    jLabel1.setToolTipText("IP address of Database Server");
    jLabel1.setHorizontalTextPosition(SwingConstants.RIGHT);
    jLabel1.setText("Server IP:");
    jLabel1.setBounds(new Rectangle(4, 14, 58, 17));
    jPanel1.setLayout(null);
    jTextField1.setText("131.120.27.60");
    jTextField1.setBounds(new Rectangle(68, 9, 245, 22));
    jTextField1.addKeyListener(new java.awt.event.KeyAdapter() {

    });

    jLabel2.setToolTipText("File to upload to Server");
    jLabel2.setHorizontalAlignment(SwingConstants.RIGHT);
    jLabel2.setText("File:");
    jLabel2.setBounds(new Rectangle(5, 41, 52, 15));
    browseButton.setToolTipText("Browse for file to load to Server");
    browseButton.setText("Browse ...");
    browseButton.setBounds(new Rectangle(314, 36, 99, 21));
    browseButton.addActionListener(new java.awt.event.ActionListener() {

        public void actionPerformed(ActionEvent e) {
            browseButton_actionPerformed(e);
        }

    });
    jTextField2.setBounds(new Rectangle(67, 36, 238, 21));
    submitButton.setToolTipText("Submit file to Server");
    submitButton.setText("SUBMIT");
    submitButton.setBounds(new Rectangle(68, 64, 81, 20));
    submitButton.addActionListener(new java.awt.event.ActionListener() {

        public void actionPerformed(ActionEvent e) {
            submitButton_actionPerformed(e);
        }

    });

    jScrollPane1.getViewport().setBackground(Color.white);
    jScrollPane1.setBounds(new Rectangle(8, 89, 432, 227));
    log.setToolTipText("Event Log");
    log.setEditable(false);
    portLabel.setToolTipText("Port Database is running on");
    portLabel.setText("Port: ");
    portLabel.setBounds(new Rectangle(325, 10, 39, 20));
    jTextFieldport.setText("80");
    jTextFieldport.setBounds(new Rectangle(360, 10, 41, 22));
    this.getContentPane().add(jPanel1, BorderLayout.CENTER);
    jPanel1.add(jTextField1, null);
    jPanel1.add(jTextField2, null);
    jPanel1.add(browseButton, null);
    jPanel1.add(jLabel1, null);
    jPanel1.add(jLabel2, null);
    jPanel1.add(submitButton, null);
    jPanel1.add(jScrollPane1, null);
    jPanel1.add(portLabel, null);
    jPanel1.add(jTextFieldport, null);
}

```

```

        jScrollPane1.getViewport().add(log, null);
    }

    //Start the applet
    public void start() {
    }

    //Stop the applet
    public void stop() {
    }

    //Destroy the applet
    public void destroy() {
    }

    //Get Applet information
    public String getAppletInfo() {
        return "Applet Information";
    }

    //Get parameter info
    public String[][] getParameterInfo() {
        return null;
    }

    //Main method
    public static void main(String[] args) {
        Applet1 applet = new Applet1();
        JFrame frame = new JFrame();
        frame.setDefaultCloseOperation(3);
        frame.setTitle("Submit Data File to Server Version: 1.1");
        frame.getContentPane().add(applet, BorderLayout.CENTER);
        applet.init();
        applet.start();
        frame.setSize(450, 350);
        Dimension d = Toolkit.getDefaultToolkit().getScreenSize();
        frame.setLocation((d.width - frame.getSize().width) / 2, (d.height -
frame.getSize().height) / 2);
        frame.setVisible(true);
    }

    void browseButton_actionPerformed(ActionEvent e) {

        int returnVal = fc.showOpenDialog(Applet1.this);

        if (returnVal == JFileChooser.FILES_ONLY) { //only let user select file
            File file = fc.getSelectedFile();
            jTextField2.setText(file.getAbsolutePath());
        }

    }

    void jTextField2_actionPerformed(ActionEvent e) {

    }

    void submitButton_actionPerformed(ActionEvent e) {

        // Action to be performed when SUBMIT is button is press

        /*
        * Opening up data File
        */
        try { //try for opening file
            log.append("Opening File: " + jTextField2.getText()+"\n");
            String fileName = jTextField2.getText();
            fileR = new FileReader(fileName);
            st = new StreamTokenizer(

```

```

        new BufferedReader(fileR));
    st.quoteChar('');    //This specified that token will be contain inside
""
    } //end try
    catch (Exception error) { //catch for opening file
        log.append(error+"\nUnable to open file\n");
    }

    /*
    * Compose the connection URL
    */

    String url = signature;    // Always starts with this

    int port = Integer.parseInt(jTextFieldport.getText());
    try{
        URL url2 = new URL("http://" + jTextField1.getText()+":"+port);

        url += url2.getHost()+":"+port + conn_etc;    // Compose the connection
URL
        log.append("URL = "+url+"\n");
        /*
        * Create an IDS JDBC Driver of the j102.sql package
        */
        log.append("Initializing IDS JDBC Driver...\n");

        IDSDriver drv = new j102.sql.IDSDriver();

        /*
        * Pick a preferred cipher-suite instead of the default.
        * Normally, you should use the default cipher-suites,
        * which means by simply calling drv.connect(url, null);
        * Refer to Section 5.3.5 of the User's Guide for detail.
        */
        Properties info = new Properties();
        /*
        * WARNING: Do not use any DH_anon_WITH_?_?_? cipher-suites
        * without fully understand the risks of "man in the middle"
        * attack. It is use here so that the example does not
        * depend on the a default public/private key pair.
        * Refer to Section 5.5 of the User's Guide for detail.
        *
        * Normally, you should use the default cipher-suites,
        * which means by simply calling drv.connect(url, null);
        */
        info.put("CipherSuite_1", "DH_anon_WITH_DES40_CBC_SHA");

        /*
        * Connect to the IDS Server using the composed URL
        */
        log.append("Connecting to IDS Server...\n");
        Connection conn = drv.connect(url, info);
//        Connection conn = drv.connect(url, null);

        ResultSet rs;
        int n;
        /*
        * Create a dummy Statement class instance
        */
        log.append("Creating SQL statement object...\n");
        Statement stmt = conn.createStatement();
        /*
        * Loop through the first three queries in the query[] array.
        * Note that the same Statement instance is re-used during
        * each iteration.
        */
        log.append("Executing SQL statements...\n");
        // Returns number of milliseconds since Jan 1, 1970
        long time = System.currentTimeMillis();
        Date currentDate = new java.sql.Date(time);

```

```

String gageName = "";
String caliResult= "";
// token should be : gageID, gageName, calibrationResult
st.nextToken();
while(st.ttype != st.TT_EOF)
{

    String gageID = st.sval;
    log.append(gageID+" ");
    st.nextToken();
    gageName = st.sval;
    st.nextToken();
    caliResult = st.sval;
    log.append("Data submitting: " + gageID + " " + gageName +
        " " + caliResult +
        " " + currentDate.toString()+"\n");

    String submitData = "INSERT INTO TABLE1 " +
        "VALUES ('" + gageID + "' , '" +
        gageName + "' , '" +
        caliResult + "' , '" +
        currentDate + "')";

    stmt.executeUpdate(submitData);
    st.nextToken();
} //close while !st.TT_EOF
try {
    fileR.close();
} catch (IOException ioerror) {
    log.append("fileR.close() unsuccessful \n");
    log.append(ioerror + "\n");
}

stmt.close();
conn.close();

log.append("Data Successfully Submitted.\n");
} //end try for submitButton_actionPerformed
catch (Throwable error2) {
    log.append(error2.toString());
}

} //end jbInit()
} //end Applet1 class

```

APPENDIX B – HTML CODE TO VIEW DATA

[illegible]

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX C – VIEWDATA JAVA APPLET

```
/*
 * This is the Java Applet for viewing calibration data from the database.
 */

package calibration;

import java.util.Properties;
import java.util.Date;
import java.net.URL;
import java.applet.*;
import java.awt.*;
import java.sql.*;

public class ViewData extends Applet implements Runnable {

    static String signature = "jdbc:ids://";
    static String conn_etc =
"/scripts/idss.dll/conn?dsn=Calibration&ssl=1&http=1";

    private TextArea textWin = null;
    private Thread theThread = null;
    private TextField status = new TextField("Please wait...");
    private Panel top = new Panel();
        private Panel bottom = new Panel();
    private TextField host = new TextField();
    private Label prompt = new Label("Enter IDS Server address: ");
    private Button runButton = new Button(" Retrieve Calibration Data ");
    private boolean firstRun = true;
        private Choice selectMe = new Choice();
    private Label instruct = new Label("Select a gage ID and press 'Get
It!' for more detailed calibration information:");
    private Button getIt = new Button("Get It!");

    // for opening a separate browser window.
    AppletContext appletContext;
    private String selectedMe; // output of getSelectedItem from selectMe.

    public void init() { // Applet initialization
        super.init();

        // Create a non-proportional font
        Font fn = new Font("Courier", Font.PLAIN, 12);
        // Create the output text window
        textWin = new TextArea();
        // Assign the fixed font to the output window
        textWin.setFont(fn);
        // Assign a layout manager for the top group
        top.setLayout(new BorderLayout());
        // Add components into the top group
        top.add("West", prompt);
        top.add("Center", host);
        top.add("East", runButton);

        Panel moreInfo = new Panel();
        moreInfo.setLayout(new BorderLayout());
        moreInfo.add("West", instruct);
        moreInfo.add("Center", selectMe);
        moreInfo.add("East", getIt);

        bottom.setLayout(new GridLayout(2, 1));
        bottom.add(moreInfo);
        bottom.add(status);
    }
}
```

```

        // Assign a layout manager for the main group
        setLayout(new BorderLayout());
        // Add components into the main group
        add("North", top);
        add("Center", textWin);
        add("South", bottom);

        appletContext = getAppletContext();
    }

    // place a border for the applet.
    public Insets getInsets() {
        return new Insets(10, 10, 10, 10);
    }

    public void start() {
        textWin.setText("");
        if (theThread == null) {
            theThread = new Thread(this);
            theThread.start();
        }
    }

    public void stop() {
        if ((theThread != null) && theThread.isAlive())
            theThread.stop();
        theThread = null;
    }

    public void run() {
        try {
            /*
             * Let's first compose the connection URL
             */

            String url = signature; // Always starts with this
            String addr = host.getText();
            if (addr.length() == 0) { // Do nothing if no address
                status.setText("Ready ok");
                if (firstRun) {
                    // Fill in the host name field
                    host.setText(getHost());
                    firstRun = false;
                }
                return;
            }
            url += addr + conn_etc; // Compose the connection URL
            /*
             * Create an IDS JDBC Driver of the j102.sql package
             */
            status.setText("Initializing IDS JDBC Driver...");
            IDSDriver drv = new j102.sql.IDSDriver();
            /*
             * Pick a preferred cipher-suite instead of the default.
             * Normally, you should use the default cipher-suites,
             * which means by simply calling drv.connect(url, null);
             * Refer to Section 5.3.5 of the User's Guide for detail.
             */
            Properties info = new Properties();
            /*
             * WARNING: Do not use any DH_anon WITH_?_?_? cipher-suites
             * without fully understand the risks of "man in the middle"
             * attack. It is use here so that the example does not
             * depend on the a default public/private key pair.

```

```

    * Refer to Section 5.5 of the User's Guide for detail.
    *
    * Normally, you should use the default cipher-suites,
    * which means by simply calling drv.connect(url, null);
    */
    info.put("CipherSuite_1", "DH_anon_WITH_DES40_CBC_SHA");
    /*
    * Connect to the IDS Server using the composed URL
    */
    status.setText("Connecting to IDS Server...");
    Connection conn = drv.connect(url, info);

    ResultSet rs;
    int n;
    /*
    * Create a dummy Statement class instance
    */
    status.setText("Creating SQL statement object...");
    Statement stmt = conn.createStatement();
    /*
    * Loop through the first three queries in the query[] array.
    * Note that the same Statement instance is re-used during
    * each iteration.
    */
    status.setText("Executing SQL statements...");

        rs = stmt.executeQuery("SELECT * FROM TABLE1");

        selectMe.removeAll();
        printQuery(rs);

    stmt.close();
    conn.close();

        status.setText("Ready ok");

    }
    catch (Throwable e) {
        // Print the exception string in the status line
        status.setText(e.toString());
    }
}

    public void openMoreInfoPage() {
        String url;
        try {
            // use AppletContext to open a new browser window
            selectedMe = selectMe.getSelectedItem();
            url = "http://" + getHost() + "/moreinfo/" + selectedMe + ".html";
            URL selectMeUrl = new URL(url);
            appletContext.showDocument(selectMeUrl, "More Info Window");
        }
        catch (Throwable e) {
            // Print the exception string in the status line
            status.setText(e.toString());
        }
    }

    // this is Java 1.0 approach to handle mouse events.
    public boolean action(Event event, Object arg) {
        // Capture the mouse click of the Run button
        if (event.target == runButton) {
            // Clear the text window
            textWin.setText("");

```

```

        run();
        return true;
    }

    else if (event.target == getIt) {
        openMoreInfoPage();
        return true;
    }

    else return super.action(event, arg);
}

private static String line = "-----";

void printQuery(ResultSet rs) throws SQLException {
    // Get the ResultSetMetaData
    ResultSetMetaData md = rs.getMetaData();

    int i, nCol = md.getColumnCount();
    // Print out the name of each returned column
    // for (i = 1; i <= nCol; ++i)
    // print(md.getColumnNames(i) + " ");
    // println();
    // Print a dividing line between column and result
    // for (i = 1; i <= nCol; ++i) {
    // s = md.getColumnNames(i);
    // Assume line is long enough for all columns
    // print(line.substring(0, s.length()) + " ");
    // }
    // println(); // Start a new line
    /*
    * Loop through each returned row in the ResultSet
    */

    String s, ss;
    int idLength = 15;
    int nameLength = 30;
    int resultLength = 8;

    s = md.getColumnNames(1);
    if (s.length() < idLength) {
        ss = s;
        for (i = 0; i < idLength - s.length(); i++) ss += " ";
    } else ss = s;
    print(ss);

    s = md.getColumnNames(2);
    if (s.length() < idLength) {
        ss = s;
        for (i = 0; i < nameLength - s.length(); i++) ss += " ";
    }
    else ss = s.substring(0, nameLength);
    print(" " + ss);

    s = md.getColumnNames(3);
    if (s.length() < resultLength) {
        ss = s;
        for (i = 0; i < resultLength - s.length(); i++) ss += " ";
    }
    else ss = s.substring(0, resultLength);
    print(" " + ss);

    s = md.getColumnNames(4);
    print(" " + s);
    println(); // Start a new line

    s = "";

```

```

        for (i = 0; i < idLength + nameLength + resultLength + 13; i++)
            s += "-";
        print(s); println();
    while (rs.next()) {
        /*
        * Loop through each column of a row
        */

        //for (i = 1; i <= nCol; ++i) {
            // Get the value of the column in String
            // s[i-1] = rs.getString(i);
        //}

        s = rs.getString(1);
        if (s.length() < idLength) {
            ss = s;
            for (i = 0; i < idLength - s.length(); i++) ss += " ";
        } else ss = s;
        print(ss);

        selectMe.add(s);

        s = rs.getString(2);
        if (s.length() < nameLength) {
            ss = s;
            for (i = 0; i < nameLength - s.length(); i++) ss += " ";
        }
        else ss = s.substring(0, nameLength);
        print(" " + ss);

        s = rs.getString(3);
        if (s.length() < resultLength) {
            ss = s;
            for (i = 0; i < resultLength - s.length(); i++) ss += " ";
        }
        else ss = s.substring(0, resultLength);
        print(" " + ss);

        s = rs.getString(4);
        if (s == null) ss = "unknown";
        else ss = s.substring(0,10);
        print(" " + ss);
    println(); // Start a new line
    }
}

/*
* Extract the origin of the running applet
*/
String getHost() {
    URL url = getCodeBase();
    int port = url.getPort();
    return url.getHost() + ':' + (port > 0 ? port : 80);
}

/*
* For less typing and better readability
*/
public void print(String s) {
    textWin.appendText(s);
}
public void println(String s) {
    textWin.appendText(s + '\n');
}
public void println() {

```

```
        textWin.appendText ("\n");  
    }  
}
```

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX D – HTX CODE TO VIEW DATA

```
<html>
<head>
<title>Calibration Result Summary</title>

<sql query=gage dsn="Calibration">
  select * from table1
</sql>

</head>

<body>

<H2>Calibration Result Summary :</H2>
<br>

<table border="1" cellpadding="6" cellspacing="0">
<tr><td><b>GAGE_ID</b></td><td><b>NOMENCLATURE</b></td><td><b>STATUS</b></td>
<td><b>DATE</b></td></tr>

<fetch query=gage>

<tr><td><a href="moreinfo/#.GAGE_ID#.html">#.GAGE_ID#</a><td>#.NOMENCLATURE#
<td>#.STATUS#<td>

<script language="JavaScript">
  var convertDate;
  convertDate = "#.DATE#";
  document.write(convertDate.substring(0,10));
</script>

</tr>
</fetch>
</table>

</body>
</html>
```


THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX E – ASP CODE TO VIEW DATA

```
<%@ Language=JavaScript %>
<!-- View data asp codes. -->
<%
/*
 * This is the asp JavaScript code for retrieving calibration data from the Access database.
 * Created on May 26, 2001.
 *
 * Modified by ... on ...
 * Modification:
 *
 */
%>
<html>
<BODY BGCOLOR="#AAFFFF">
<head>
<title>Calibration Result Summary</title>
</head>
<body>

<H2>Calibration Result Summary:</H2>
<br>
<%
// standard statements for ODBC database connection.

var DSN, Conn, sql, rs;

DSN = "DSN=Calibration";
Conn = Server.CreateObject("ADODB.Connection");
Conn.Open(DSN);

sql = "select * from table1";
rs = Server.CreateObject("ADODB.RecordSet");
rs.Open(sql, Conn);
var i;

// retrieve the database field names and keep in an array for later reference.
field_array = new Array();
for (i=0; i < rs.fields.count; i++) {
field_array[i] = rs.fields(i).name;
}
// table setup things.
Response.Write("<table border='1' cellpadding='6' cellspacing='0'>");
Response.Write("<tr>");

// put the field names in the first row of the table.
for (i=0; i < rs.fields.count; i++) {
Response.Write("<td><b>" + field_array[i] + "</b>");
}
Response.Write("</tr>");

// fill the table with data. mm/day/year is extracted from the Date() object.
var db_date, month;
```

```

while (! (rs.EOF)) {
db_date = new Date(String(rs.fields.item(field_array[3])));
month = db_date.getMonth() + 1; // getMonth() returns 0 to 11 for January to December.

// this portion may be improved with a for-loop to make it more general.
Response.Write("<tr><td>" + String(rs.fields.item(field_array[0])) + "<td>" +
String(rs.fields.item(field_array[1])) + "<td>" + String(rs.fields.item(field_array[2])) +
"<td>" + month + "/" + db_date.getDate() + "/" + db_date.getFullYear() + "</tr>");
rs.move(1);
} // end of while loop

// close the database connection.
rs.Close();
Conn.Close();
// close the table
Response.Write("</table>");
%>

<br>
<br>
<a href="viewdata.asp" > <b> View submitted data again. </b>
<br>
<br>
<a href="submitdata.asp" > <b> Submit a file to the database. </b>
</body>
</html>

```

APPENDIX F – ASP JAVASCRIPT CODE TO SUBMIT DATA

```
<%@ Language=JavaScript %>

<!-- Submit data. -->

<%

/*
 * This is the asp JavaScripts codes for submitting calibration data from the Access database.
 * It uploads a file containing calibration data to a directory on the web server, and then
 * transfers data fields in the file to the calibration result Access database, which is used
 * to retrieve data for viewing purpose.
 *
 * All the server-side scripting is carried out by SubmitFileScript.asp
 *
 * Created on May 26, 2001.
 *
 * Modified by Xiaoping Yun on May 27, 2001
 * Modification: to accept wide range of data field delimiters.
 * The code uses Javascripts 1.2 Regular Expressions.
 */

%>

<html>

<head>
<title>Submitting Calibration Results </title>
</head>

<body bgcolor="#AFFFFFF">

<h2>Submit a calibration result file to the database server: </h2>

<FORM METHOD="POST" ENCTYPE="multipart/form-data" ACTION="SubmitFileScript.asp">
<table width=80%>
<tr> <td width=15% align="left"> File Name:</td> <td width=85%> <INPUT TYPE=FILE SIZE=40
NAME="FILE1"> </td>
<tr> <td> Delimiter*: </td> <td> <INPUT TYPE=TEXT SIZE=40 NAME="Delimiter"></td>
<tr> <td> </td> <td> <INPUT TYPE=SUBMIT VALUE="Submit"> </td>
</table>
</FORM>

<p> <font size=1 color="#0000AA" >
<br>
* Note: The default delimiter is assumed to be
two-charater string
<font color=red> &;
</font> (i.e., ampersand followed by semicolon) if nothing is entered
in the delimiter space above. In this case, the data file has the following format: <br> <br>
GAGE ID &; Gage Name &; Pass/fail Status &; Date <br> <br>
Any single characters (except /) or strings can be used for the delimiter.
```

Enclosing data fields by double quote " " is not implemented at this time.

 </p>

</body>

</html>

APPENDIX G – SUBMITFILESSCRIPT.ASP

```
<%@ Language=JavaScript %>

<HTML>
<BODY BGCOLOR="#AAFFFF">

<%

/*
 * This is the script file for uploading files and for submitting data to
 * the Access database.
 *
 * Created on May 26, 2001.
 *
 * Modified by ... on ...
 * Modification:
 *
 */

var Upload, File, Count, file_to_open, DelimiterItem;
Upload = Server.CreateObject("Persits.Upload.1");
Upload.OverwriteFiles = false;

// first save the file on the hard disk of the server.
// it should be replaced with user login name.
Count = Upload.Save("c:\\users/home/web")

// for testing purpose
// Response.Write(Count + " files submitted. <br>");

// get the name of the file just saved. This name may be different from the
// original file name if there are duplicates in the directory.
File = Upload.Files(1);
DelimiterItem = Upload.Form(1);

file_to_open = File.Path;

fso = new ActiveXObject("Scripting.FileSystemObject");

// standard statements for ODBC database connection.
var DSN, Conn, sql;
DSN = "DSN=Calibration";
Conn = Server.CreateObject("ADODB.Connection");
Conn.Open(DSN);

var one_line, fields;
var delimiter_char;
```

```

/*
 * If no delimiter is entered or entered delimiter is one or more space, tab, new line,
 * the default delimiter "&," is assumed. Otherwise, take the entered the delimiter.
 */
if ((DelimiterItem.value.length == 0) || (DelimiterItem.value == "\s+")) {
    delimiter_char = "&,"; // default delimiters.
}
else {
    delimiter_char = DelimiterItem.value;
}

// for testing purpose.
// Response.Write("The current delimiter is:" + delimiter_char + "<br>");

if (fso.FileExists(file_to_open)) {

file_stream = fso.OpenTextFile(file_to_open);

while (! file_stream.AtEndOfStream ) {
    one_line = file_stream.ReadLine();

    // for testing purpose.
    // Response.Write(one_line + delimiter_char + "<br>");

    /*
     * Parse records from each line of submitted calibration data.
     * In the following, the record fields are assumed to be separated by semicolon.
     * Other separator would be fine too.
     * Escape charaters are not implemented initially.
     */

    // Split the line into an array of fields separated by the delimiter.
    fields = one_line.split(delimiter_char);

    // for testing purpose.
    // Response.Write("number of fields" + fields.length + "<br>");

    // if the data files are not properly written, it may causes error
    // when writing to the database with more or less number of fields.

    // Compose the SQL INSERT statement.
    // If the line does not have enough fields, ignore it.
    // If the line has more than the number of total fields, ignore the extra fields.
    number_of_max_field = 4; // current number.
    if ( fields.length >= number_of_max_field ) {
        sql = "INSERT INTO TABLE1 " + "VALUES (" +
            fields[0] + " , " +
            fields[1] + " , " +
            fields[2] + " , " +
            fields[3] + ")";

        // Insert the data into database.
        Conn.Execute(sql);
    }
}
}

```

```

        } // enf of if loop

    } // end of while-loop.

    // close the database connection.
    Conn.Close();

    // close the file.
    file_stream.close();

    Response.Write("The following file has been successfully submitted to the database server: <br>");
    Response.Write(File.OriginalPath + "<br>");

    } // end of if file exists check

    else
    {
        Response.Write("<p> The file does not exist or access is denied.");
    } // end of else

%>

<br>
<br>
<a href="submitdata.asp" > <b> Submit another file. </b>

<br>
<br>

<a href="viewdata.asp" > <b> View all submitted data. </b>

</BODY>
</HTML>

```


THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF REFERENCES

1. TechFest, "TechFest Ethernet Technical Summary."
[<http://www.techfest.com/networking/lan/ethernet1.htm#1.1>]. October 2000.
2. Clarinet System, "The EthIR LAN Family."
[<http://www.clarinetsys.com/site/products-page/family-page/product-overview.htm>]. October 2000.
3. IRLAN, "The Principles of IR Diffused Technology."
[<http://www.irlan.co.il/principles.html>]. October 2000.
4. AEI Wireless Communications, "E100 Fast Ethernet Series Laser Bit Documentation."
[<http://www.aeiwireless.com/html/fastethernet100mbps.html>]. November 2000.
5. Cisco Systems, "Cisco Aironet Antennas and Accessories Data Sheet."
[http://www.cisco.com/warp/public/cc/pd/witc/ao340ap/prodlit/airoa_ds.htm]. March 2001.
6. Kaplan, Gadi, "Ethernet's Winning Ways," *IEEE Spectrum*, vol. 38, Is. 1, pp. 113-115, January 2001.
7. Shotsberger, Paul, and Ron Vetter, "Teaching and Learning in the Wireless Classroom," *Computer*, vol. 34, no. 3, pp. 110-111, March 2001.
8. Quarterman, John S., "The History of the Internet and the Matrix."
[<http://members.iquest.net/~jswartz/jks/professional/history.html>]. March 2001
9. Institute of Electrical and Electronics Engineers, *IEEE Standard 802.11a-1999 – Supplement to IEEE Standard 802.11-1999*, Institute of Electrical and Electronics Engineers, Inc., New York, NY, 1999.
10. HIPERLAN2, HIPERLAN2 Global Forum.
[<http://www.hiperlan2.com/web/>]. March 2001.
11. Federal Communication Commission, "FCC Radio Spectrum Home Page – Spectrum Management in the United States."
[<http://www.fcc.gov/oet/spectrum/>]. March 2001.
12. Spread Spectrum Scene, "The ABC of Spread Spectrum – A Tutorial."
[<http://sss-mag.com/ss.html#tutorial>]. March 2001.

13. Katz, Randy H., "CS294-7: Radio Propagation."
[<http://sss-mag.com/pdf/1propagation.pdf>]. April 2001.
14. Naval Air Warfare Center, "EW and Radar Systems Engineering Handbook – Antennas Radiation Patterns."
[<http://ewhdbks.mugu.navy.mil/RADIAPAT.HTM>]. March 2001.
15. Bing, Benny, *High-Speed Wireless ATM and LANs*, Artech House Publishers, Boston, MA, 2000.
16. DesBrisay, Grey presentation on Basics of Orthogonal Frequency Division Multiplexing, April 12, 2001.
17. Woerner, Brian D., "EcpE 5654 – Digital Communications Project – PI/4 DQPSK with match filters."
[<http://www.mprg.ee.vt.edu/people/woerner/dc/projects/p7.pdf>]. April 2001.
18. Cisco Systems, "White Paper – Overcoming Multipath in Non-Line-of-Sight High-Speed Microwave Communication Links."
[http://www.cisco.com/warp/public/cc/pd/witc/wt2700/mulpt_wp.htm]. May 2001.
19. Beaumont, Chris, "Secret Communication System – The Fascinating Story of Lamarr/Antheil Spread-Spectrum Paten,"
[<http://www.ncafe.com/chris/pat2/index.html>]. April 2001.
20. Matthews, Mark M., *Analysis of Radio Frequency Components for Shipboard Wireless Networks*, Master's Thesis, Naval Postgraduate School, Monterey, California, December 1999.
21. McConnel, Richard J., *Testing and Evaluation of Shipboard Wireless Network Components*, Master's Thesis, Naval Postgraduate School, Monterey, California, March 2000.
22. Institute of Electrical and Electronics Engineers, *IEEE Standard 802.11-1997*, Institute of Electrical and Electronics Engineers, Inc., New York, NY, 1999.
23. Conover, Joel, "802.11a: Making Space for Speed." *Network Computing*, January 8, 2001.
[<http://www.networkcomputing.com/1201/1201ws1.html>]. March 2001.
24. RSA Laboratories, "What is RC4?"
[<http://www.rsasecurity.com/rsalabs/faq/3-6-3.html>]. April 2001.

25. Borisov, N., Goldberg, I., and Wagner, D. "Intercepting Mobile Communications: The Insecurity of 802.11."
[<http://www.isaac.cs.berkeley.edu/isaac/wep-draft.pdf>]. May 2001.
26. Cisco Systems. "Cisco Aironet 350 Series Wireless LAN Security Overview – The Growth of Wireless LANs."
[http://www.cisco.com/warp/public/cc/pd/witc/ao350ap/prodlit/a350w_ov.htm]. May 2001.
27. Institute of Electrical and Electronics Engineers, "IEEE P802.11 Wireless LANs – 802.11 Task Group E Security Subgroup Minutes," Institute of Electrical and Electronics Engineers, Inc., New York, NY, March 12-15, 2001.
28. Linksys, "PCM 100 – EtherFast 10/100 Integrated PC Card."
[<http://www.linksys.com/products/product.asp?prid=100&grid=11>]. February 2001.
29. 3Com, "Product Details – 3Com OfficeConnect Dual Speed Hub 8."
[http://www.3com.com/products/en_US/detail.jsp?tab=features&pathtype=purchase&sku=3C16750B-US]. February 2001.
30. Cisco System. "Cisco Aironet 340 Series LAN Wireless Solutions."
[http://www.cisco.com/warp/public/cc/pd/witc/ao340ap/prodlit/airo_pg.ppt] January 2001.
31. Walden, Cole, McGraph, and Young. "GPTE Semi-Automated High Pressure Calibrator Manual," NAVSEA Metrology R&D Program.
32. IDS Software. "IDS Server's User Guide," Rev. 3.2.2, February 2000.
33. Sun Microsystems. "JDBC Data Access API."
[<http://java.sun.com/products/jdbc/>]. May 2001.
34. Sun Microsystems. "JavaServer Pages – Dynamically Generate Web Content."
[<http://java.sun.com/products/jsp/>]. May 2001.
35. Fields, D. K. and Kolb, M. A., *Web Development with JavaServer Pages*, Manning Publications Co., 2000.
36. AspUpload. "About AspUpload."
[<http://www.aspupload.com/about.html>]. June 2001.
37. US Navy. "Measure 21."
[<http://www.nalda.navy.mil/measure/>]. June 2001.

THIS PAGE INTENTIONALLY LEFT BLANK

INITIAL DISTRIBUTION LIST

1.	Defense Technical Information Center	2
	8725 John J. Kingman Road, Suite 0944	
	Ft. Belvoir, VA 22060-6218	
2.	Dudley Knox Library	2
	Naval Postgraduate School	
	411 Dyer Road	
	Monterey, CA 93943-5101	
3.	Chairman, Code EC.....	1
	Department of Electrical and Computer Engineering	
	Naval Postgraduate School	
	Monterey, CA 93943-5121	
4.	Professor Xiaoping Yun, Code EC/YX.....	6
	Department of Electrical and Computer Engineering	
	Naval Postgraduate School	
	Monterey, CA 93943-5121	
5.	Professor John McEachen, Code EC/MJ	1
	Department of Electrical and Computer Engineering	
	Naval Postgraduate School	
	Monterey, CA 93943-5121	
6.	Commandant (G-CIT).....	1
	2100 Second Street SW	
	Washington, DC 20593-0001	
7.	Commandant (G-SRF)	1
	2100 Second Street SW	
	Washington, DC 20593-0001	
8.	LT Tung Ly	1
	4404 Amador Rd.	
	Fremont, CA 94538-1202	